

零点起航



—— 计算机编程系列教材

Visual FoxPro 7.0



基 础 教 程

◎零点工作室 孙杰 白伟青 高冬梅 等编著



机械工业出版社
CHINA MACHINE PRESS

零点起航 —— 计算机编程系列教材

Visual FoxPro 7.0 基础教程

零点工作室 孙杰 白伟青 高冬梅 等编著



机械工业出版社

Visual FoxPro 7.0 是 Microsoft 公司推出的新版本的数据库应用系统。本书通过大量实例深入浅出地介绍了 Visual FoxPro 7.0 的使用方法,使读者可以方便地掌握该软件的各项功能。全书共分 12 章,主要内容包括概论、Visual FoxPro 的语言基础、表的制作、数据库的操作、查询与视图、项目、程序设计基础、面向对象程序设计基础、表单、菜单和工具栏设计、报表及应用程序的发布等。此外,书中还给出了有关章节的实验、Visual FoxPro 7.0 的常用函数、数据库操作命令、常用事件和方法等内容。

本书内容全面、深入,既适合初、中级读者和大专院校师生,也适合开发人员作为参考资料。

图书在版编目(CIP)数据

Visual FoxPro 7.0 基础教程/孙杰等编著. —北京:机械工业出版社, 2005.1

零点起航——计算机编程系列教材

ISBN 7-111-15575-0

I. V... II. 孙... III. 关系数据库—数据库管理系统, Visual FoxPro 7.0—程序设计—教材 IV. TP311.138

中国版本图书馆 CIP 数据核字(2004)第 114356 号

机械工业出版社(北京市百万庄大街 22 号 邮政编码 100037)

责任编辑:商红云

封面设计:陈 沛 责任印制:李 妍

北京机工印刷厂印刷·新华书店北京发行所发行

2005 年 1 月第 1 版第 1 次印刷

787mm×1092mm 1/16·16.5 印张·394 千字

定价:23.00 元

凡购本书,如有缺页、倒页、脱页,由本社发行部调换

本社购书热线电话(010) 68993821、88379646

封面无防伪标均为盗版

系列教材序言

随着计算机技术的飞速发展,各种应用件也在社会和领域得到了广泛应用,软件的设计和开发也成为日常工作中的一部分。对于从事计算机方面工作的人员,掌握一种或几种编程工具是其必须具备的专业技能,也是胜任工作的基本条件。同时,很多高校学生、电脑爱好者也希望自己能掌握基本的软件设计方法,以满足未来工作和学习的需要。

零点起航——计算机编程系列教材选择了目前常用的编程软件,涵盖了应用程序设计、数据库开发和网络编程开发工具,着眼于大专院校教师、学生和工程技术人员的自学和培训。整套由7本书组成:

- (1)《Visual Basic.NET 基础教程》
- (2)《Visual C++.NET 基础教程》
- (3)《PowerBuilder 9.0 基础教程》
- (4)《Java 2 基础教程》
- (5)《C++Builder 6.0 基础教程》
- (6)《Delphi 基础教程》
- (7)《Visual FoxPro 7.0 基础教程》

整套教材本着从零末始的思想,从基础培训的角度入手,在内容的选限和章节的设置上充分考虑了初学者的实际需要,力求简明清晰、通俗易懂。在详细讲解软件功能和用法的同时,引导读者练习一些针对性、实用性很强的程序实例,以加深内容的理解。在每章的最后,都附带了一些习题,通过对这些习题的思考和练习,读者可以对该章所学内容有更深刻的认识。该套教材的作者都是长期从事计算机教学和软件设计的高校教师和专业人员,具有丰富的理论和实践经验,在写作过程中融入了多年的经验和体会,为初学者提出了许多有益的建议。

零点起航——计算机编程系列教材中的各教材自成体系,读者可以根据自己的实际需要选择。

希望这套丛书对您的学习、工作和生活有所帮助。

零点工作室网站: <http://www.zerobook.net>

主编邮箱: guandianxhu@qdcnc.com

零点工作室

2004年6月

前 言

Microsoft 公司新推出的 Visual FoxPro 7.0 具有强大的功能，它以面向对象编程为基础，拥有丰富和完整的开发工具，且简单易学、开发成本低，备受广大用户的欢迎。Visual FoxPro 7.0 完全兼容其早期版本 Visual FoxPro 6.0，并在此基础上，强化了可视化编程工具，提高了支持 XML 的能力，而且增加了近 50 命令与函数。

本书深入浅出地介绍了 Visual FoxPro 7.0 的使用方法。书中采用大量实例介绍了该软件的各项功能，初学者可以参照这些实例方便地掌握各个知识点。

书中以“学生信息管理系统”为例贯穿全文，便于读者系统全面地掌握该软件的使用方法，并以期通过本书的学习使读者能够完成小型数据库应用系统的开发。

在本书的编写过程中，力求以新颖和实用的角度介绍 Visual FoxPro 7.0。在内容取舍上，针对高等院校本、专科学生的特点和实际教学的需要，进行了合理安排，力求使学习者可以在较短的时间内基本掌握 Visual FoxPro 7.0 的主要功能和使用方法。

全书共分 12 章，此外还包含实验和附录部分。书中各章节的编写分工如下：第 1、6、9 章由孙杰编写，第 2、3 章由高冬梅编写，第 4、10 章由白伟青编写，第 5、8 章由杨金龙编写，第 7、11 章和附录部分由田更编写，第 12 章由解统燕编写。参加编写的还有魏爱民、王美兴、韩亮、高雪霞、刘春秋和刘萍等。全书由孙杰提出编写指导思想，由孙杰和白伟青完成最后的统稿工作。

在本书的编写过程中，尽管我们尽了很大努力，但不足之处在所难免，恳请读者在使用本书过程中予以指正。

编者

2004 年 6 月

目 录

系列教材序言

前言

第 1 章 概论	1	第 5 章 查询与视图	90
1.1 数据库基本概念	1	5.1 查询的创建与使用	90
1.2 关系数据库	4	5.2 视图的创建和使用	102
1.3 Visual FoxPro 7.0 的特点、环境 配置和安装	8	5.3 小结	108
1.4 小结	14	5.4 习题	108
1.5 习题	14	第 6 章 项目	110
第 2 章 Visual Foxpro 的语言基础	14	6.1 项目的创建	110
2.1 数据类型和数据存储	14	6.2 项目管理器	112
2.2 运算符和表达式	18	6.3 应用程序框架	116
2.3 命令和函数	21	6.4 小结	118
2.4 小结	29	6.5 习题	118
2.5 习题	29	第 7 章 程序设计基础	119
第 3 章 表的制作	30	7.1 程序文件的建立、编辑与调用	119
3.1 表的创建	30	7.2 程序的基本结构	123
3.2 向表中输入数据	36	7.3 子程序与过程	135
3.3 数据的显示与修改	40	7.4 程序设计中应注意的问题	143
3.4 数据的删除	41	7.5 小结	146
3.5 小结	42	7.6 习题	146
3.6 习题	43	第 8 章 面向对象程序设计基础	150
第 4 章 数据库的操作	44	8.1 面向对象基本概念	151
4.1 数据库的设计	44	8.2 VFP 中的对象和类	154
4.2 数据库的创建	48	8.3 小结	163
4.3 数据库表的操作	56	8.4 习题	163
4.4 索引	71	第 9 章 表单	164
4.5 数据库表的关联	81	9.1 表单的创建	164
4.6 小结	88	9.2 表单控件的使用	173
4.7 习题	88	9.3 小结	190
		9.4 习题	190

第 10 章 菜单和工具栏设计	191	12.3 小结	226
10.1 菜单的创建	191	12.4 习题	226
10.2 工具栏的设计	204	实验	227
10.3 小结	209	实验一 数据库操作	227
10.4 习题	209	实验二 查询与视图	227
第 11 章 报表	210	实验三 程序设计	229
11.1 创建报表	210	实验四 类的创建和使用	231
11.2 修改报表	216	实验五 制作表单	231
11.3 小结	217	附录	232
11.4 习题	217	附录 A 常用函数一览表	232
第 12 章 应用程序的发布	218	附录 B 数据库操作命令一览表	241
12.1 连编项目文件	218	附录 C Visual FoxPro 常用事件	251
12.2 应用程序的发布	222	附录 D Visual FoxPro 常用方法	253
		参考文献	255

第1章 概 论

计算机以其自身所具有的极高的运算速度、巨大的存储容量和精确的算术和逻辑运算等特点,被广泛应用于各行各业。特别是在信息极大丰富的当今社会,信息的采集和加工更是离不开计算机技术的深入发展和广泛使用。数据库技术是信息处理的核心技术,掌握与其相关的一些基本概念是学习 Visual FoxPro 的前提和基础。本章主要讲解了数据库、数据模型和数据库管理系统等基本概念,并对关系数据库作了一些阐述,最后介绍了一下 Visual FoxPro 7.0 的特点、环境配置和安装。

1.1 数据库基本概念

数据管理和数据处理是计算机系统的最基本的支撑技术。数据库技术作为一种进行数据管理和数据处理的典型技术,已被广泛应用。下面简单介绍一下数据库的一些基本概念。

1.1.1 信息与数据

为了认识客观世界和进行彼此交流,人们需要各种信息。信息是客观事物的属性的反映,是事物之间相互联系、相互作用的状态描述。与信息相关的是数据。数据是信息存在的一种形式,是信息的载体。数据有一定的格式,例如一名学生的信息可以用如下格式的数据来描述:“赵强,3039757,男,1979,山东,市场营销,1997”。该数据包含的信息为:一名叫赵强的男学生,学号为3039757,1979年出生,山东人,1997年考入市场营销专业学习。

数据常见的形式有:

- 数值型数据:如年龄、身高、数量等。
- 字符型数据:如姓名、班级、专业等。
- 逻辑型数据:如是否已婚、是否毕业等。
- 多媒体型数据:如声音、图形、图像、视频等。

信息和数据是两个不同的概念,信息是有价值的信息,它以数据为载体,依靠数据来完成信息的传播。而数据却不一定具有价值。

1.1.2 数据模型

现实世界中事物总是彼此联系的,由于数据是描述客观事物的载体,因此数据与数据之间必然存在一定的联系,这种联系可以用数据模型来描述。由此,我们把数据模型定义为反映事物间联系的数据组织的结构和形式。数据模型包含两个方面的内容。其一为数据的静态特性,即数据的基本结构、数据间的联系和数据的约束;其二为数据的动态特性,即定义在数据上的操作。例如,学生记录中包含姓名、学号、性别、出生年月等字段,每

个字段有数据类型和长度约束，对学生记录可以进行添加、查询、删除等操作。
常用的数据模型有如下三种：

1. 层次模型

层次模型表示的数据间的联系是一种从属关系。可以用树来描述层次模型，树根为最
高层，其所处位置最高；除树根外，所有结点都应有唯一的父结点。如图 1-1 所示，一个
学院包含多个系，每个系又包含多个班。

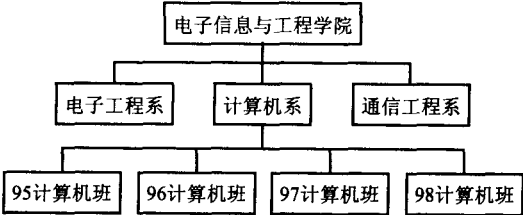


图 1-1 层次模型示例

2. 网状模型

网状模型中，一个结点可以有多个父结点，
两个结点间可以有多种联系。如图 1-2 所示，T2
的父结点是 T1 和 T3，P1 和 P4 之间的联系可以
是 P4→P1,也可以是 P1→P2→P3→P4。

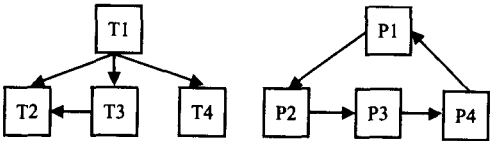


图 1-2 网状模型示例

3. 关系模型

关系模型是以集合论中的关系概念为基础发展起来的，它常把数据看成是一个二维表，
称这个表为关系。如表 1-1 所示，表中每一列是一个属性：每一行称为一个元组，即一条
记录。可以采用传统的集合运算（如，并、交、差）和专门的关系运算（如，投影、选择、
连接）来完成数据的处理。

表 1-1 关系模型示例

学 号	姓 名	性 别	出生年月	专 业
30297021	袁永红	女	1980, 3	计算机
30298011	纪祥师	男	1979, 5	计算机
30297220	江志强	男	1981, 11	物业管理
30299134	刘小江	男	1979, 2	国际贸易

1.1.3 数据库系统

数据库系统的基本结构如图 1-3 所示。图
中包含数据库、数据库管理系统、数据库管理
员和多个应用程序，下面分别进行介绍。

1. 数据库

数据库是数据的汇集，它们以一定的组织
形式存于存储介质上，存储介质一般采用磁
盘。数据库将相关的数据组织起来，以实现数

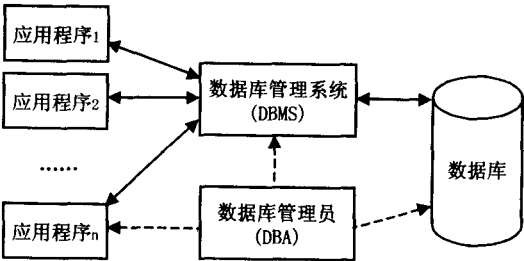


图 1-3 数据库系统

据的共享和多用户访问，并实现数据与应用程序的彼此独立。数据库的类型由其数据模型决定。如果数据库中的数据采用层次模型进行组织，则该数据库称为层次型数据库；如果数据库中的数据采用网状模型进行组织，则该数据库称为网状型数据库；如果数据库中的数据采用关系模型进行组织，则该数据库称为关系型数据库。Visual FoxPro 是采用关系模型来组织数据的，由此可知它的数据库是关系型数据库。

2. 数据库管理系统

数据库管理系统 (database management system, 简称 DBMS) 是管理数据库的软件，它实现数据库系统的各种功能，是数据库系统的核心。数据库管理系统对数据库系统的功能和性能有决定性的影响。目前市场上绝大多数已商品化的 DBMS 是关系型 DBMS。DBMS 最基本的功能是正确、安全、可靠地执行数据库语言语句。与高级语言一样，DBMS 有编译和解释两种实现方法。

DBMS 一般按不同的应用程序需求提供多种用户接口，例如交互式 SQL 接口、嵌入式 SQL 接口、表格语言接口、自然语言查询接口等。

随着计算机技术的不断发展，DBMS 的系统结构也在不断变化，其 4 种主要的系统结构为：

- 分时系统环境下的集中式数据库系统结构。
- 网络环境下的客户/服务器结构。
- 物理上分布、逻辑上集中的分布式数据库结构。
- 物理上分布、逻辑上分布的分布式数据库结构。

3. 数据库管理员

由于数据库是实现数据的共享的，必然需要对数据库进行规划、设计、协调、维护和管理，负责完成这些工作的人或集体称为数据库管理员 (database administrator, 简称 DBA)。DBA 是拥有最高特权的数据库用户。DBA 既要有一定的专业知识，又要有一定的管理知识；他（们）既要了解系统，又要非常熟悉用户的需求。

一般来说，DBA 有下列任务：

- 充分了解数据库的设计。
- 组织完成数据库的建立和安装。
- 了解用户需求，解决有关技术问题；指导用户正确使用数据库。
- 进行数据的完整性维护。
- 保证数据库的安全。
- 对数据库进行适当的调整，以保证运行性能。
- 在必要的时候对数据库进行重构。
- 制定有关规章制度。

4. 应用程序

用户要访问的数据是以一定的物理形式存放在数据库中的，如果让用户直接访问这种物理形式的数据，势必要求用户要掌握许多技术细节，显然这是不可能实现的。为此，可以编制相应的数据库应用程序，为用户提供友好的访问界面。通过执行应用程序，用户只

需提出需要什么数据，而不必关心如何获取这些数据。不同的应用程序与 DBMS 连接时，需要不同的接口，现在的 DBMS 可提供包括图形接口在内的多种高级用户接口，大大方便了用户对数据库的访问。

一般的，常用的数据库应用程序有以下体系结构：

- 基于单机的数据库应用程序。这种结构中，应用程序和数据库是在同一台计算机中。故此，这种结构无法实现数据的多用户共享。
- 基于客户机/服务器模式的数据库应用程序：这种结构中，应用程序和数据库一般是在不同的计算机中。应用程序运行在客户机上，数据库则运行在服务器上。
- 基于浏览器/服务器模式的数据库应用程序：这种结构中，客户机上只运行浏览器程序，数据库运行在数据库服务器上，二者之间应该有 1 台或多台应用服务器，上面集中了控制数据的逻辑代码（或系统企业逻辑），所以是数据关系控制的核心。其目的是为不同的客户端应用提供数据共享，同时可以保证数据的一致性。

1.2 关系数据库

关系数据库是由若干个依照关系模型设计的数据表文件组成的集合。也就是说，关系数据库包含若干张二维数据表。每一个数据表有若干条记录组成，每一条记录是由不同字段属性值构成。每一个数据表以一个独立的表文件形式存在，同一个数据库中不允许有重名的表文件名。一个数据库中的表文件是彼此相关的，这种相关性恰好表现了数据表所描述的事物之间的联系。

1.2.1 基本术语

1. 字段

字段就是客观事物的“属性”，是构成数据库表的“数据项”。从二维表格结构来看，是表的“列”。例如表 1-1 中的学号、姓名、性别和出生年月等。

2. 记录

字段的有序集合称为记录。在关系模型中，记录称为元组。从二维表格结构来看，记录是表的“行”。例如表 1-1 中的一条记录为：“30297021、袁永红、女、1980，3、计算机”。

3. 表

记录的集合称为表。在关系模型角度来说，表也可以称为是关系。例如表 1-1 所示的为学生信息记录集合。

4. 关键字

关键字是能够唯一确定记录的字段或字段的集合。有时候记录中可能存在多个关键字，我们从中取出一个作为主关键字，简称主键。例如表 1-2 所示的学生信息表中字段“学号”为该表的主关键字。如果一个表中引用的字段或字段的集合不是它的关键字，恰好是其他表的关键字，则称之为该表的外关键字，简称外键。例如表 1-3 所示的成绩表中字段“学号”是学生信息表的主关键字，但不是成绩表的关键字，此时，称“学号”为成绩表的外关键字。

表 1-2 学生信息表

学 号	姓 名	性 别	出生年月	专 业
30297021	袁永红	女	1980, 3	计算机
30298011	纪祥师	男	1979, 5	计算机
30299134	刘小江	男	1979, 2	国际贸易
30297220	江志强	男	1981, 11	物业管理

表 1-3 课程信息表

课 程 号	课 程 名	学 分	总 学 时	开课学期
21020721	操作系统	2	80	2000-2001, 一
21010202	英语	4	160	2000-2001, 一
21010201	高等数学	4	160	2000-2001, 一

1.2.2 关系数据库的特点

关系数据库是由若干数据表构成的，而这些数据表是依照关系模型设计完成的，数据表之间既相互联系，又彼此独立，从而使关系数据库具有极大的优越性。

关系数据库的特点如下：

- 支持复杂的数据结构，数据冗余度很小。
- 数据库的数据和应用程序在逻辑结构和物理存储方式上是相互独立的，方便了用户的使用。
- 数据库的数据具有共享性，这样数据库可为多个用户服务。
- 关系数据库提供多种控制功能，可以保证数据的安全性、完整性、一致性和多用户同时访问的并发控制。

1.2.3 关系操作

1. 投影

投影是对一个数据表中的字段进行筛选，并按要求重新组合。例如从学生信息表中选择“学号”和“姓名”构成一个新表，如表 1-5 所示。

2. 选择

选择是从一个表中选择符合条件的记录集，并构成一个新的表。例如从学生信息表中选择出专业为“计算机”的学生，如表 1-6 所示。

3. 连接

连接是在两个表中指定的字段集上选取满足一定条件的记录，组成新的表。例如从学生信息表中的字段学号、姓名、专业，与成绩表（表 1-4）中的字段学号、课程号、成绩，连接生成新表，如表 1-7 所示。

表 1-4 成绩表

学 号	课 程 号	成 绩
30297021	21020721	80
30298011	21010202	90
30299134	21010201	79
30297220	21010202	81

表 1-5 投影示例表

学 号	姓 名
30297021	袁永红
30298011	纪祥师
30299134	刘小江
30297220	江志强

表 1-6 选择示例表

学 号	姓 名	性 别	出生年月	专 业
30297021	袁永红	女	1980, 3	计算机
30298011	纪样师	男	1979, 5	计算机

表 1-7 连接示例

学 号	姓 名	专 业	课 程 号	成 绩
30297021	袁永红	计算机	21020721	80
30298011	纪样师	计算机	21010202	90
30297220	江志强	物业管理	21010201	81

1.2.4 关系完整性

数据库的完整性是指数据的正确性和相容性。例如一个人的年龄应该在 0~150 之间, 不能出现负数或太大的数值。另外, 表与表之间存在着一定的联系和约束, 如成绩表中出现的字段“学号”应该是学生信息表中存在的“学号”, 否则会出现错误。

关系模型的完整性规则是对关系的某种约束条件。关系模型中有 3 种基本的完整性约束:

- 实体完整性: 一个数据表(或关系)通常对应现实世界中的一个实体集。例如学生信息表对应学生集合。现实世界中的实体是可以区分的, 同样的数据表中的数据也是可区分的, 其区分的依据是主关键字。因此, 主关键字中的所有属性的值不能取空值。例如在学生信息表中, 若学号取空值, 就无法区分和说明学生的情况。

- 参照完整性: 实体之间一般存在着某种联系, 这种联系同样可以用数据表(或关系)来描述。这样就必然存在表与表之间的引用。例如表 1-3 成绩表中引用了表 1-2 学生信息表中的主关键字“学号”。也就是说, 成绩表中“学号”属性取值要参照学生信息表中的主关键字“学号”的取值。

- 域完整性约束: 数据表中的每一个属性(或字段、列)对应着一个值的集合, 作为其可以取值的范围, 称为该属性的域。属性值应该是域中的值, 以及一个属性能否为空都是域完整性约束的内容。域完整性约束是最简单和最基本的约束, 一般 DBMS 中都有域完整性约束检查功能。

实体完整性、参照完整性和域完整性约束是关系数据模型的 3 个最基本、最普遍的完整性约束。不同的 DBMS 也有一些其他的完整性约束。为保证数据库的一致性和正确性, 必须使数据库中的数据满足完整性约束。在实际应用中, 完整性约束的检查具体由用户完成, 还是由 DBMS 来完成, 要根据实际采用的技术来定。

1.2.5 关系的规范化 (Relation Normalization)

关系的规范化的基本思想是逐步消除数据依赖关系中不合适的部分, 使反映现实世界中同一实体的数据达到有效的分离。下面分别介绍一下第一范式、第二范式和第三范式的基本内容。

1. 第一范式

在一个关系中, 消除重复字段, 且各字段都是不可分的基本数据项, 则这个关系符合第一范式。例如表 1-7 中不存在重复字段, 因此表 1-7 符合第一范式。如果该表中存在两个或多个相同的字段, 则该表就不符合第一范式。

2. 第二范式

首先一个关系应属于第一范式，而且关系中所有非主属性应完全依赖于关键字段，则这个关系符合第二范式。如表 1-8 所示的成绩表，主关键字为（学号、课程号），这个关系属于第一范式。表中存在下列关系：

（学号、课程号）→课程名，课程号→课程名，即非主属性“课程名”不完全依赖于关键字段（学号、课程号）。

因此，表 1-8 所示的成绩表不符合第二范式。只要将表中的字段“课程名”去掉，变成表 1-4 所示的成绩表，由于表 1-4 符合第一范式，且关系中所有非主属性完全依赖于关键字段（学号、课程号），因此表 1-4 符合第二范式。

3. 第三范式

首先一个关系应属于第二范式，而且关系中所有非主属性都直接依赖于关键字段，则这个关系符合第二范式。如表 1-9 所示的学生信息表中，假设各个系中不存在重复专业，关键字段为“学号”，显然该表符合第二范式。表中存在下列关系：

学号→系所（即，一个学号可以唯一确定一个系所，或者说系所依赖于学号），系所→专业，同时不存在系所→学号（或者说不存在学号依赖于系所），即有学号→系所→专业，显然“专业”传递依赖于“学号”，而不是直接依赖于“学号”。

因此，表 1-9 所示的学生信息表不符合第三范式，将表中字段“系所”去掉，得到表 1-2 所示的学生信息表，此时关系中的所有非主属性都直接依赖于关键字段“学号”，因此该表符合第三范式。

表 1-8 成绩表

学 号	课 程 号	课 程 名	成 绩
30297021	21020721	操作系统	80
30297021	21010201	高等数学	82
30297021	21010202	英语	78
30298011	21010202	英语	90
30299134	21010201	高等数学	79
30297220	21010202	英语	81

表 1-9 学生信息表

学 号	姓 名	性 别	出生年月	系 所	专 业
30297021	袁永红	女	1980, 3	计算机系	计算机
30298011	纪祥师	男	1979, 5	计算机系	计算机
30299134	刘小江	男	1979, 2	国际贸易系	国际贸易
30297220	江志强	男	1981, 11	管理系	物业管理

按照规范化标准设计的数据库通常比非规范化的数据库占有的数据空间小，数据表之间的关系也更加清晰和明确，从而使数据的访问变得更加容易。但实际应用中，并非数据库所用范式标准越高越好，因为所用范式标准太高会使数据表之间的联系过于复杂，反而会降低数据库的运行性能。一般的，采用第三范式进行数据库的数据规范即可。

1.2.6 表的关联

数据库中表的关联关系有一对一、一对多和多对多关系。下面分别介绍一下这几种关系。

1. 一对一关系

一对一关系，即在 2 个数据表中选择一个相同字段作为关键字段，在一个表中该字段为主关键字，其值唯一，在另外一个表中该字段为外关键字，其值也唯一。在这种关系中，每一个表中的一个记录最多只与相关表中的一个记录相关联。比如，表 1-9 所示的学生信息表中“学号”为主关键字，表 1-10 所示的住宿情况表中也包含“学号”，但它不是主关键字，一个学生只能有一个住所，因此住宿情况表中应该只有一条该学生的有效记录，也就是说学生信息表和住宿情况表构成了一对一的关联关系。

表 1-10 住宿情况表

学 号	房 间 号	床 位 号	备 注
30297021	21010121	01	
30298011	21040322	02	
30299134	21050210	04	
30297220	21030202	02	

2. 一对多关系

一对多关系，即在 2 个数据表中选择一个相同字段作为关键字段，在一个表中该字段为主关键字，其值唯一，在另外一个表中该字段为外关键字，其值是重复的。在这种关系中，主表中的每一个记录与相关表中的多个记录相关联。比如，表 1-9 所示的学生信息表中“学号”为主关键字，表 1-8 成绩表中也包含“学号”，但它不是主关键字，一个学生可以学习多门课程，因此成绩表中应该有多条该学生的成绩记录，也就是说学生信息表和成绩表构成了一对多的关联关系。

3. 多对多关系

在这种关系中，每一个表中的一个记录在相关表中都可能有多条记录与其匹配。比如，一个教师可以教授多门课程，同样的一门课程也可以由多个教师教授，这样在课程和教师之间就构成了一种多对多的关联关系。

1.3 Visual FoxPro 7.0 的特点、环境配置和安装

Microsoft 公司新推出 Visual FoxPro 7.0 完全兼容其早期版本 Visual FoxPro 6.0，并在此基础上，强化了可视化编程工具，提高了支持 XML 的能力，而且增加了近 50 命令与函数。下面简单介绍一下 Visual FoxPro 7.0 的特点、环境配置和安装。

1.3.1 Visual FoxPro 7.0 的特点

与以往版本相比，Visual FoxPro 7.0 为我们提供了更加强大的数据库管理系统 (DBMS)，它具有更快的速度、更强大的功能和更高的灵活性。此外 Visual FoxPro 7.0 为我们提供了面向对象和基于事件的新一代编程语言，使我们可以更加快捷地编制和修改应用程序。在下面的内容中我们将介绍一下 Visual FoxPro 7.0 的主要特点。

1. 拥有快速应用程序开发工具

利用 Visual FoxPro 的向导、工具栏和设计器等开发工具，可以让我们快速构建应用程序系统。如图 1-4 和图 1-5 所示，利用 Visual FoxPro 提供的向导可以快速构建应用程序。

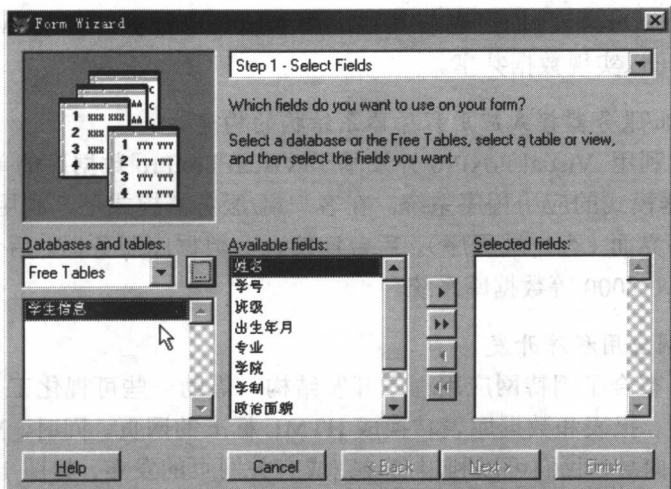


图 1-4 表单向导

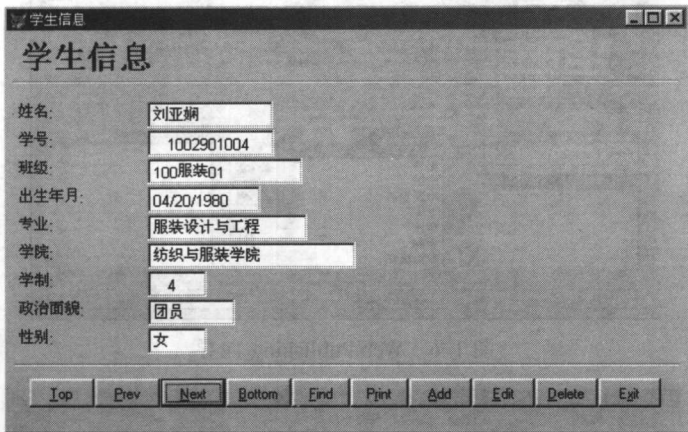


图 1-5 应用表单向导构建的表单应用程序

2. 具有更加强大的编程能力

Visual FoxPro 7.0 编程能力比以前的版本更加强大。面向对象编程和事件结构的采用使编程更加容易。Visual FoxPro 在数据库中，配合专有的复合索引技术，只需要非常少的时间即可完成大量的数据存取，此种技术被称为 Rushmore 技术。由于采用了该技术，使 Visual FoxPro 应用系统的执行速度大大提高。

3. 创建用户界面无需编程

Visual FoxPro 提供强大的设计工具，我们只需使用很少的代码或根本不使用代码就可以创建用户界面。

4. 支持多人共同开发一个应用系统

当多个程序员同时进行一个应用系统的开发时，Visual FoxPro 为此提供了数据库同时访问机制。此外，我们可以利用项目管理器跟踪程序源代码的改变，以避免出现错误。

5. 支持与其他应用系统交互

利用 Visual FoxPro，我们可以与其他应用系统共享数据。如与 Microsoft Excel 和 Microsoft Word 等可以实现数据共享。

6. 支持客户机/服务器模式应用程序体系结构的构建

我们不但可以利用 Visual FoxPro 开发基于 Visual FoxPro 数据库的单机系统，还可以开发客户机/服务器模式的应用程序系统。在客户机/服务器模式下，利用 Visual FoxPro 编写前台的应用程序界面（客户端程序），后台连接一个数据库服务器。后台的数据库服务器可以采用 SQL Server2000 等数据库系统。

7. 支持因特网应用程序开发

Visual FoxPro 整合了因特网应用程序开发结构，借助一些可视化工具，我们可以轻松地直接将窗体界面、报表和数据库表转换成 HTML 标准的网页。同时，Visual FoxPro 提供了 Web Publishing 向导，我们可以利用向导完成有关网页的发布，如图 1-6 和图 1-7 所示。

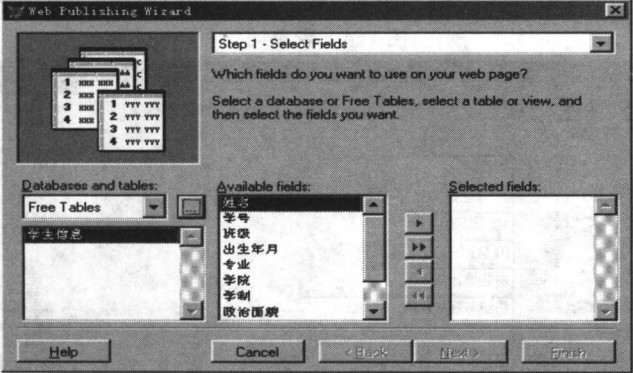


图 1-6 Web Publishing 向导

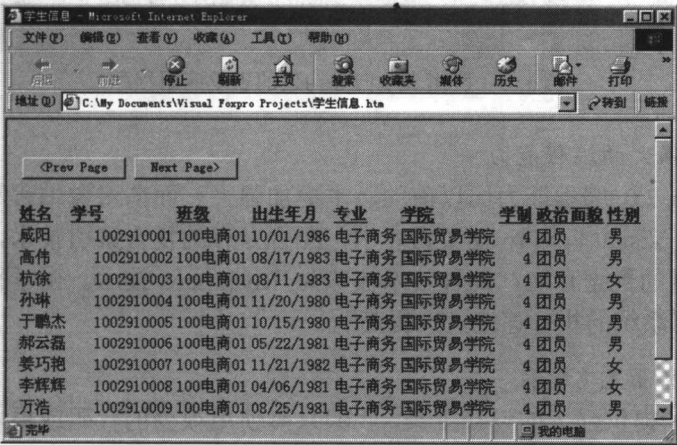


图 1-7 利用 Web Publishing 向导发布的 web 页面