



中文版 Office 2000

王 川 阮兴平 编著
于平剑 何建波

VBA

一册通

人民邮电出版社

中文版 Office 2000 VBA 一册通

王 川 阮兴平 编著
于平剑 何建波

人 民 邮 电 出 版 社

内 容 提 要

本书主要介绍 Microsoft Office 2000 VBA (Visual Basic for Applications) 的使用方法和操作技巧。使用 VBA 可以方便地操作 Office 2000, 深入开发 Office 2000 的强大功能, 使 Office 办公系列软件更加高效地工作。全书共分为 16 章, 每一章都通过具体的实例和有序的操作步骤来指导读者循序渐进地进行实际练习, 同时还通过大量的提示、注释和图片来帮助读者快速掌握所学内容。

本书内容由浅入深, 叙述清晰, 通俗易懂, 适合广大学习和使用 Office 2000 VBA 的用户阅读, 也可供大中专院校师生学习和参考。

中文版 Office 2000 VBA 一册通

- ◆ 编 著 王 川 阮兴平 于平剑 何建波
责任编辑 马 嘉
- ◆ 人民邮电出版社出版发行 北京市崇文区夕照寺街 14 号
北京密云春雷印刷厂印刷
新华书店总店北京发行所经销
- ◆ 开本: 787×1092 1/16
印张: 21.5
字数: 536 千字
印数: 1~6 000 册
- 2000 年 1 月第 1 版
2000 年 1 月北京第 1 次印刷

ISBN 7-115-07678-2/TP·1030

定价: 33.00 元

前 言

随着信息技术的不断发展和计算机的迅速普及，越来越多的计算机用户使用各种应用软件和编程工具来完成日常的工作。中文版 Microsoft Office 2000 正式发布以后，得到了广大计算机用户的普遍认同和好评，用户可以利用它的强大功能来完成几乎所有的日常工作。Microsoft Office 2000 中的套件 Visual Basic for Application (VBA) 则为用户更加深入地应用 Office 提供了支持。

通常一提到编程，您可能马上就会想到 C 语言、C++ 语言、Visual C++ 等，以及各种烦琐的规定和指针等，这使很多不太熟练的编程用户感到困扰和吃力。相对而言，Microsoft Office 2000 中的 Visual Basic 则简单易学，用户不用像使用 C 语言那样必须从最底层开始编程，Microsoft Office 2000 中的 Visual Basic 提供的各种控件和交互式的编程方式可以让一个从没有学过编程的人很快就能编出合格的程序，完成自己的任务。Visual Basic 是 Windows 环境下开发应用软件的一种通用程序设计语言，具有功能强大、简便易用等特点。在 Office 2000 中文版中提供的宏语言 VBA 是 Visual Basic 的一个子集，它可以广泛地应用于微软公司开发的各种软件中。使用好 VBA 能够更方便地操作 Office 2000，更好地控制 Office 2000，更深入地开发 Office 2000 的强大功能，使用户能够方便快捷地定制和扩展 Office 2000 功能系统，让 Office 2000 办公系列软件能更好地协调工作。

本书以通俗易懂的语言从 VBA 的各个方面来讲述 Microsoft Office 2000 中 Visual Basic 的各种应用。本书第 1~8 章介绍了 Office 2000 中的 VBA 的特点、功能、对象及其属性和方法，接着介绍了 VBA 的窗体和控件，第 9~15 章具体介绍了 Office 2000 的各个组件中的 VBA 的应用，比如 Word 2000、Excel 2000、Access 2000、PowerPoint 2000 以及 Outlook 2000 中的 Visual Basic 应用，第 16 章讲述了 Office 2000 中各组件的 VBA 综合应用的实例。希望本书的内容有助于广大读者掌握 Microsoft Office 2000 中的 Visual Basic 应用，并对用户提高计算机应用水平和工作效率有所帮助。

参加本书编写工作的有王川、阮兴平、于平剑、何剑波、郑文锋、张涛、李志超、钟剑、张伟、阮子轩、周佳农、郑凯、周云龙、郭锋、曾颖峰、邓志刚等，全书的审校工作由阮兴平负责完成。

由于时间仓促，加上作者水平有限，书中的疏漏之处在所难免，希望广大读者批评指正。

作 者

目 录

第一章 快速浏览 Office 2000 编程	1
1.1 Microsoft Office 2000 与 VBA 的关系	1
1.1.1 VBA 与 Visual Basic 的联系和区别	1
1.1.2 打开 Visual Basic 编辑器	1
1.2 VBA 功能简介	4
1.3 VBA 对象的特征	7
1.3.1 属性	7
1.3.2 方法	7
1.3.3 事件	8
1.4 VBA 工程的结构	8
1.5 Windows 的组件模型	10
第二章 Microsoft Office 2000 的对象模型	12
2.1 什么是对象模型	12
2.1.1 原始对象	12
2.1.2 中央对象和全局对象	13
2.1.3 操作对象的属性	15
2.1.4 什么是对象集合	16
2.2 属性、事件和方法	17
2.2.1 属性	17
2.2.2 事件	20
2.2.3 方法	21
2.3 获取有关对象、方法和属性的帮助	24
2.3.1 宏录制器	24
2.3.2 自动列表成员	25
2.3.3 自动显示快速信息	26
2.3.4 对象浏览器	27
第三章 VBA 的符号系统和程序设计基础	30
3.1 Visual Basic 的数据类型	30
3.1.1 变量命名规则	30
3.1.2 数据类型种类	30
3.1.3 变量声明	31
3.1.4 常量声明	32
3.1.5 变量或常量的范围层	33
3.2 编写 VBA 代码	34
3.2.1 声明语句	34
3.2.2 赋值语句	34

3.2.3	可执行语句	35
3.2.4	编程技巧与注意事项	36
3.3	使用条件子句	39
3.3.1	使用 If 条件子句	39
3.3.2	使用 Select 条件子句	41
3.4	使用循环语句	41
3.4.1	Do...Loop 循环语句结构	42
3.4.2	For...Next 循环语句结构	44
3.4.3	For Each...Next 循环语句结构	45
3.5	调试代码	46
3.5.1	使用断点	46
3.5.2	使用立即窗口	47
3.5.3	本地窗口	48
3.5.4	监视窗口	49
第四章	窗体控件和窗体模块	50
4.1	建立一个用户窗体	50
4.1.1	设计窗体样式	50
4.1.2	设置控件属性	52
4.1.3	编制控件代码	52
4.1.4	运行程序	53
4.2	什么是控件	54
4.3	窗体模块和 VBA 工作区	59
4.3.1	窗体和容器	60
4.3.2	VBA 建立的事件过程框架	61
4.4	控件的通用事件、方法和属性	61
4.4.1	通用事件	62
4.4.2	通用方法	68
4.4.3	通用属性	69
第五章	将 ActiveX 控件应用于 VBA	72
5.1	设计用户的功能对话框一	72
5.2	设计用户的功能对话框二	88
5.3	将控件加入应用程序	97
5.3.1	了解 Office 应用程序控件工具箱	98
5.3.2	设计模式与运行模式	99
第六章	Office 2000 应用程序间的信息交换	101
6.1	创建新对象	101
6.1.1	如何使用 CreateObject 函数	101
6.1.2	使用 CreateObject 函数创建新的应用实例	102
6.1.3	采用严格的变量定义方式	105
6.1.4	使用 New 关键字	108

6.2 引用已有对象	108
6.2.1 如何使用 GetObject 函数	109
6.2.2 使用 GetObject 函数引用已有对象	110
6.2.3 释放对象变量	113
6.3 应用程序交换信息实例	113
6.3.1 应用程序 Word 与 PowerPoint 间的信息交换	114
6.3.2 应用程序 Word 与 Outlook 间的信息交换	119
第七章 创建用户菜单和工具栏	122
7.1 创建用户工具栏和菜单栏	124
7.1.1 创建用户工具栏	124
7.1.2 创建自定义菜单	129
7.1.3 添加菜单项	132
7.1.4 添加下拉控件工具按钮	135
7.2 为工具栏添加操作	138
7.3 清除界面对象	140
7.3.1 删除菜单项	140
7.3.2 删除菜单	142
7.3.3 删除工具按钮	142
7.3.4 删除工具栏	143
第八章 Office 2000 对象特征及应用	145
8.1 创建文件查找工具	146
8.1.1 创建用户自定义窗体	147
8.1.2 为自定义窗体添加功能	150
8.1.3 连接自定义对话框和菜单命令	154
8.2 创建用户的 Office 2000 助手	156
8.2.1 选择 Office 助手	157
8.2.2 创建助手气球	157
8.2.3 装饰助手气球	160
8.2.4 连接自定义对话框和帮助	161
8.3 检索文件属性	163
8.3.1 检索文档属性	164
8.3.2 设置文档属性	166
8.4 其它对象	166
8.4.1 HTMLProject 对象	166
8.4.2 COMAddIns 集合对象	168
8.4.3 LanguageSettings 对象	168
8.4.4 AnswerWizard 集合对象	169
8.4.5 Scripts 集合对象	170
8.4.6 WebPageFonts 集合对象	172
第九章 理解 Office 2000 数据库	173

9.1 理解数据库	173
9.1.1 数据库的构成	173
9.1.2 平面文件数据库与关系数据库	175
9.1.3 典型的数据库	176
9.1.4 数据库管理系统	176
9.1.5 数据访问对象	179
9.2 使用 SQL 进行数据库操作	180
9.2.1 SQL 简介	180
9.2.2 DAO 中数据库的符号表示	183
第十章 Word 2000 对象库	185
10.1 宏	185
10.2 Word 2000 对象	189
10.2.1 Word 2000 的事件	191
10.2.2 使用对象属性	193
10.2.3 使用对象方法	194
10.3 常用对象	195
10.3.1 Documents 类集合对象	195
10.3.2 Range 对象	199
10.3.3 Selection 对象	203
10.3.4 Paragraph 对象	208
10.3.5 Template 对象	210
10.4 应用实例	211
10.4.1 保存所有打开着的文档	211
10.4.2 建立一个备忘录	213
10.4.3 使用 Ontime 来设置程序运行间隔	214
第十一章 Excel 2000 对象库	217
11.1 在 Excel 2000 中记录宏	217
11.2 Excel 2000 对象	218
11.2.1 使用对象属性	219
11.2.2 使用对象方法	220
11.3 常用对象	221
11.3.1 Workbook 对象	221
11.3.2 Worksheet 对象	226
11.4 Excel 2000 的常用事件	230
11.4.1 Activate 事件	231
11.4.2 Open 事件	232
11.4.3 BeforeClose 事件	233
11.4.4 Deactivate 事件	234
11.4.5 Change 事件	234
11.4.6 BeforePrint 事件	234

11.4.7 NewWorkbook 事件	235
11.4.8 WindowResize 事件	235
11.4.9 OnKey 方法	236
11.4.10 OnTime 方法	236
11.5 应用实例	237
11.5.1 使用 OnTime 来设置程序运行间隔	237
11.5.2 保存所有打开着的工作簿	238
第十二章 Access 2000 的事务自动化	241
12.1 Access 的工作环境	241
12.2 使用 VBA	244
12.2.1 Access 的集合、对象、属性、方法及事件	244
12.2.2 使用对象属性	246
12.2.3 使用对象方法	247
12.3 常用对象	249
12.3.1 Control 对象	249
12.3.2 DoCmd 对象	250
12.3.3 Report 对象	251
12.3.4 Module 对象	252
12.3.5 Screen 对象	253
12.3.6 Form 对象	254
第十三章 PowerPoint 2000 的对象库	256
13.1 使用宏	257
13.2 使用 VBA	258
13.2.1 PowerPoint 的对象、属性和方法	258
13.2.2 使用对象属性	259
13.2.3 使用对象方法	260
13.3 常用对象	261
13.3.1 Application 对象	261
13.3.2 Presentations 集合对象	263
13.3.3 DocumentWindows 集合对象	264
13.3.4 Selection 对象	265
13.3.5 Shape 对象	267
第十四章 Outlook 2000 的特殊性	269
14.1 Outlook 2000 的脚本编辑器	269
14.1.1 VBScript 与 VBA 的区别	269
14.1.2 VBScript 在 Outlook 2000 中的作用	271
14.1.3 VBScript 的指令子句	272
14.2 Outlook 2000 窗体设计器	274
14.2.1 显示 Outlook 窗体设计器	274
14.2.2 显示或隐藏 Outlook 窗体的页面	276

14.2.3 对窗体页面重命名	276
14.3 使用控件设计窗体	277
14.3.1 在自定义 Outlook 窗体中添加控件	277
14.3.2 在标签页 Questions 上添加控件	279
14.4 使用 VBScript 编写代码	281
14.4.1 添加 Item_Open 事件过程	281
14.4.2 添加 Questions 页面控制代码	283
14.4.3 发送窗体	284
14.5 自定义窗体的保存、发布和管理	286
14.5.1 将窗体保存为文件或模板	286
14.5.2 将窗体发布在窗体程序库	287
14.5.3 为窗体设置密码和自定义信息	287
第十五章 集成化办公解决方案实例	288
15.1 创建 Access 数据库	289
15.1.1 建立数据表	289
15.1.2 设计 Access 窗体	290
15.1.3 编写窗体控制代码	291
15.1.4 创建 modMain 模块	292
15.2 从数据库中读取数据	294
15.3 创建 Excel 工作表和图表	296
15.3.1 创建 Excel 工作表	296
15.3.2 向 Excel 工作表添加条目	297
15.3.3 创建 Excel 图表	298
15.4 创建 Word 报告	301
15.5 创建 PowerPoint 演示文稿	304
15.6 用 Outlook 发送邮件	306
15.6.1 创建 Outlook 邮件	307
15.6.2 运行结果	308
第十六章 创建应用程序的加载项	309
16.1 创建 Word 加载项	309
16.1.1 创建加载项	309
16.1.2 创建用户自定义菜单	316
16.1.3 装载 Word 加载项	320
16.1.4 下载 Word 加载项	321
16.2 创建 Excel 加载宏	321
16.2.1 创建 Excel 加载宏	321
16.2.2 在 Excel 应用程序中添加自定义菜单	323
16.2.3 Excel 加载宏的装载和卸载	326
16.3 创建 PowerPoint 加载宏	326
16.4 为工程设置密码保护	332

第一章 快速浏览 Office 2000 编程

1.1 Microsoft Office 2000 与 VBA 的关系

VBA (Visual Basic for Applications) 是由微软开发的用于定制和扩展 Office 2000 功能的系统, 是功能强大的 Office 2000 套件的一个部分。VBA 能给 Office 2000 应用程序提供一种方法让用户可以完成 Office 2000 本身所不具备的功能, 并且按照自己的习惯对 Office 2000 进行定制, 制作自己习惯的工具栏, 菜单栏以及常用的自动加载功能等等。

VBA 是插入到 Office 2000 应用程序中的一个 Windows 构件, 并在不同的层次上与它们共存。VBA 与 Office 2000 应用程序 Word 2000、Excel 2000、Access 2000、PowerPoint 2000 以及 Outlook 2000 存在着紧密的联系, 它实际上是一个解释器, 将用户的程序指令作为自己独立的 Windows 进程来执行。VBA 解释器是从一个 VBA 模块文件中调用它的指令, 解释这些指令的含义, 并马上执行。

那么一个 VBA 程序或模块能影响 Office 2000 应用程序运行的原因是什么呢? 这是因为 VBA 和 Office 2000 应用程序之间有一个内部的联络物, 这就是“对象库”。它们通过定义 Office 2000 的数据部分和操作基本规则提供的通用语言来彼此对话。VBA 提供给用户改变和补充 Office 2000 的语法, 并且为与之协同工作的应用程序的对象库提供了大量的词汇。

1.1.1 VBA 与 Visual Basic 的联系和区别

VBA 的编辑器提供的工具和图形用户界面与 Microsoft Visual Basic 6.0 版程序设计系统的开发环境是一样的, 但是, Visual Basic 6.0 版提供了更多更强大的高级开发工具, 这样, 可以创建基于 Microsoft Windows 操作系统的高级程序和其它 Windows 程序部件。VBA 与 Microsoft Visual Basic 6.0 是不同的, 它通过应用 Visual Basic 的编辑器和 Office 2000 的应用程序, 像 Word、Excel 和 PowerPoint 等, 以创建 Office 2000 特殊的用户应用程序, 这些程序可以保存在 Office 2000 文档中或以加载项的文件形式单独保存。

Microsoft Word 2000、Excel 2000 和 PowerPoint 2000 都提供了相同的集成开发环境。Visual Basic 编辑器包含了编写 Visual Basic 程序和创建解决方案所需的全部程序设计工具。例如, 在 Word 2000 中可以切换到 Visual Basic 编辑器窗口, 这与从 Excel 中切换的方法是完全一样的。下面我们将介绍如何打开 Visual Basic 编辑器。

1.1.2 打开 Visual Basic 编辑器

在工具菜单中指向宏, 然后单击 Visual Basic 编辑器, 如图 1-1 所示。这样用户可以看到在屏幕上弹出了 Visual Basic 编辑器的界面, 如图 1-2 所示。

需指出的是, 在 Word 2000、Excel 2000 和 PowerPoint 2000 中, 它们所包含的 Visual Basic

编辑器是完全相同的。这样，用户有可能因为应用的需要，同时需打开三个 Visual Basic 编辑器，其中每一个均与包含它的应用程序相关，当应用程序关闭时，与其对应的 Visual Basic 编辑器也相应地自动关闭。

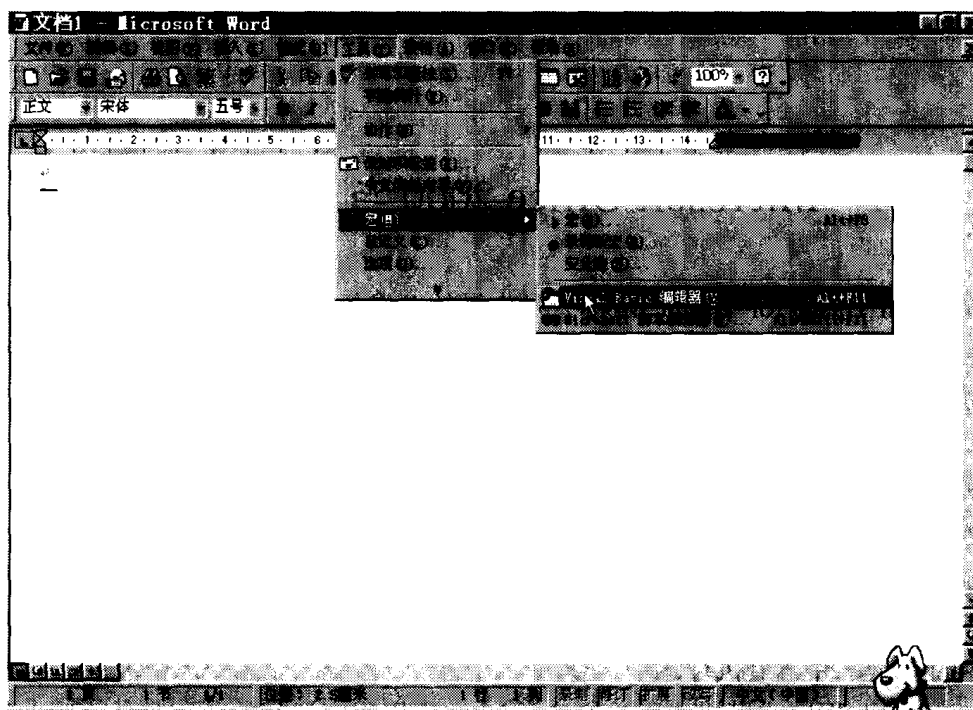


图 1-1 打开 Visual Basic 编辑器

Access 2000 和 Outlook 2000 中的 VBA 与 Word 2000、Excel 2000 和 PowerPoint 2000 中的 VBA 是不同的，它们自身提供了类似于 Visual Basic 编辑器的编辑环境，但为了使 Office 家族中各应用程序之间的协调与合作，Microsoft 将 Visual Basic 编辑器集成在所有的 Office 应用程序中。这样，我们可以用多种方式启动 Visual Basic 编辑器，如在 Access 2000 中，VBA 位于表的前一部分。用户不需要使用一个单独的 Visual Basic 编辑器来为 Access 2000 产生源代码，而是像图 1-3 那样，将用户的代码模块与 Access 2000 的窗体、数据表和计划共享主工作区。

Outlook 2000 没有提供任何高级的 Visual Basic 开发工具以及 Word 2000、Excel 2000 和 PowerPoint 2000 提供的开发环境。用户使用 Outlook 2000 时，可以通过 Visual Basic 脚本，即 Visual Basic 程序设计语言的一个子集来应用邮件、时间计划和桌面信息管理服务，同时通过 Outlook 窗体设计程序提供强大的窗体设计能力。

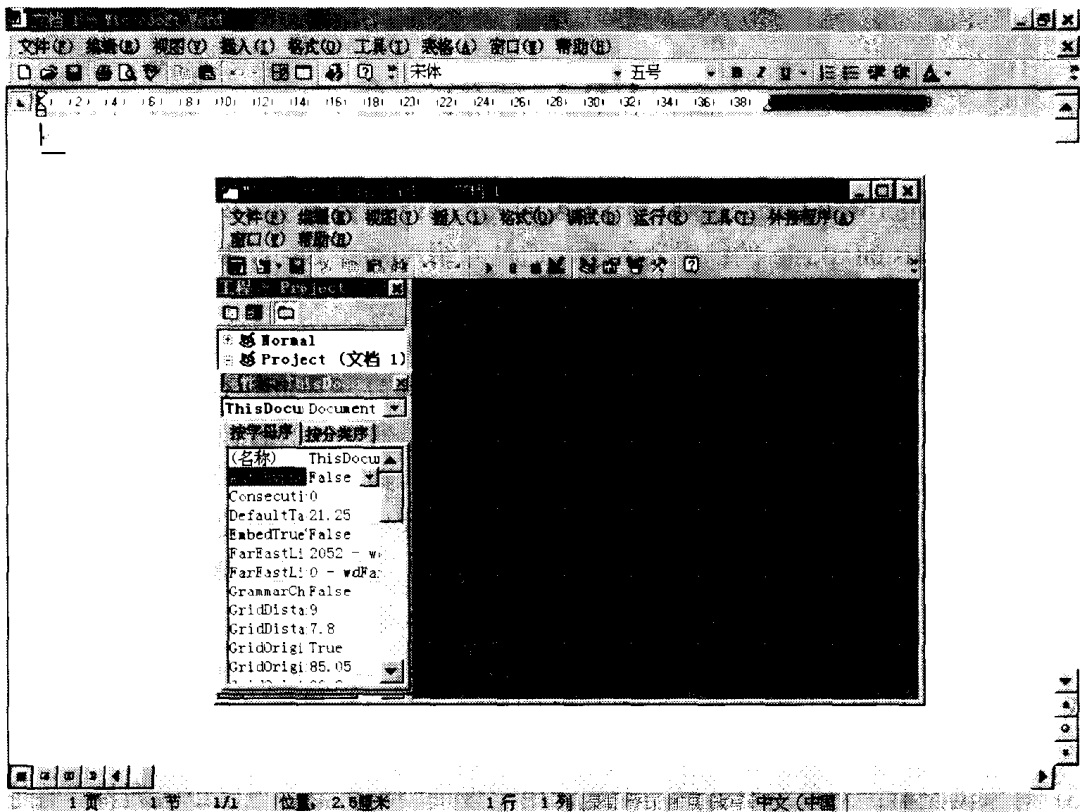


图 1-2 Visual Basic 编辑器界面

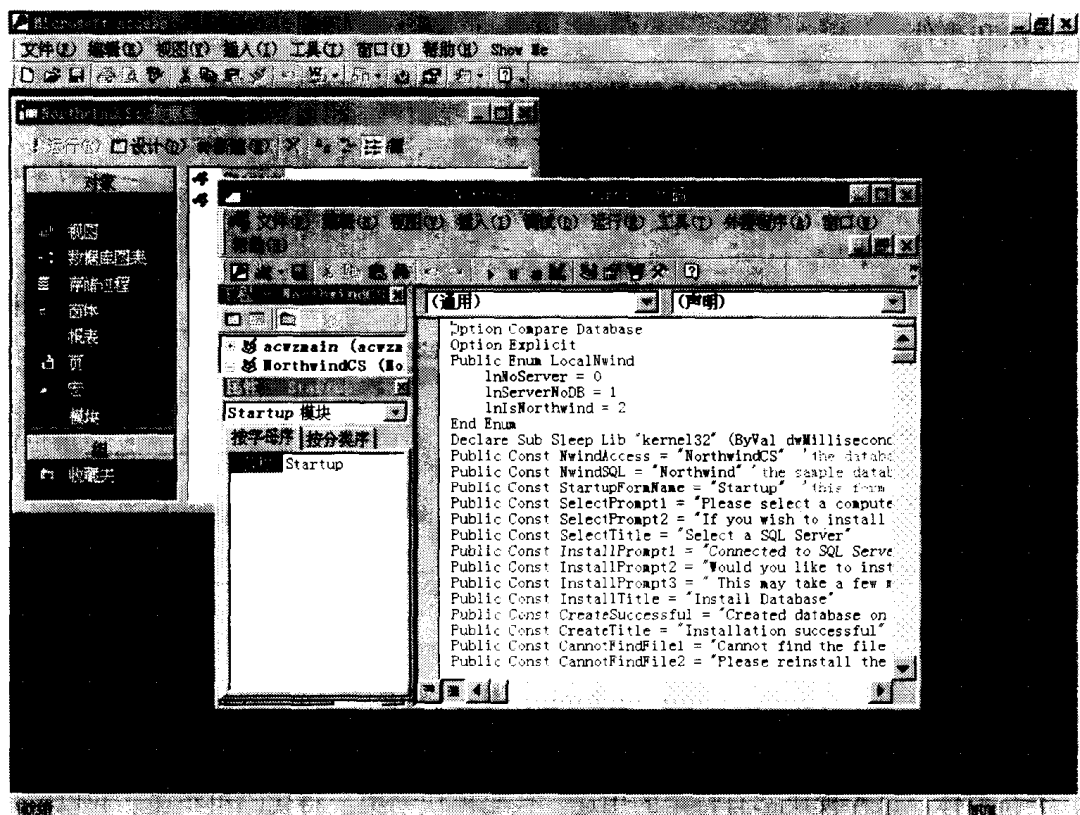


图 1-3 Access 2000 的编辑窗体

1.2 VBA 功能简介

前面介绍了 Microsoft Office 2000 与 VBA 的关系,下面将介绍 VBA 的主要功能。

首先我们从 Visual Basic 的编辑器入手,通过介绍 Visual Basic 编辑器的各个组成部分和它们的功能来分析 VBA 的功能。

Visual Basic 编辑器提供了一系列的高级程序设计和开发工具,用户在打开 Visual Basic 编辑器后,在视图菜单中可以看到如下的几个主要的工具窗口。如图 1-4 所示。

(1) 工程资源管理器:该窗口显示了一个分级的工程列表和所有被工程引用的或包含的条目。如果用户对 Windows 中的资源管理器熟悉的话,则不会对工程的结构和性能感到陌生。工程资源管理器中的每一个工程与用户打开的应用程序中的对象是一一相关的。比如,当用户打开一个 Word 2000 文档时,在工程资源管理器中就有一个 VBA 工程和它相关联。这个 VBA 工程的信息条目可以是任意数量的代码模块或用户窗体。

(2) 属性窗口:和通常的可视化编程工具类似,该窗口中显示的信息是代码模块、用户窗体或 ActiveX 控件的属性列表。左边是属性的名称,右边是相应的属性值。属性的顺序是按照字母序或者是分类序的顺序排列的。

(3) 对象浏览器:对象浏览器的功能是让用户查询浏览对象、方法或者属性。在它的帮助下,用户不必再去记忆庞大的 VBA 库、类以及成员,只需在有问题的时候在对象浏览器的帮助下找到由 ActiveX 控件、Word、Excel、PowerPoint 等应用程序所提供的浏览对象的属性、方法和事件。

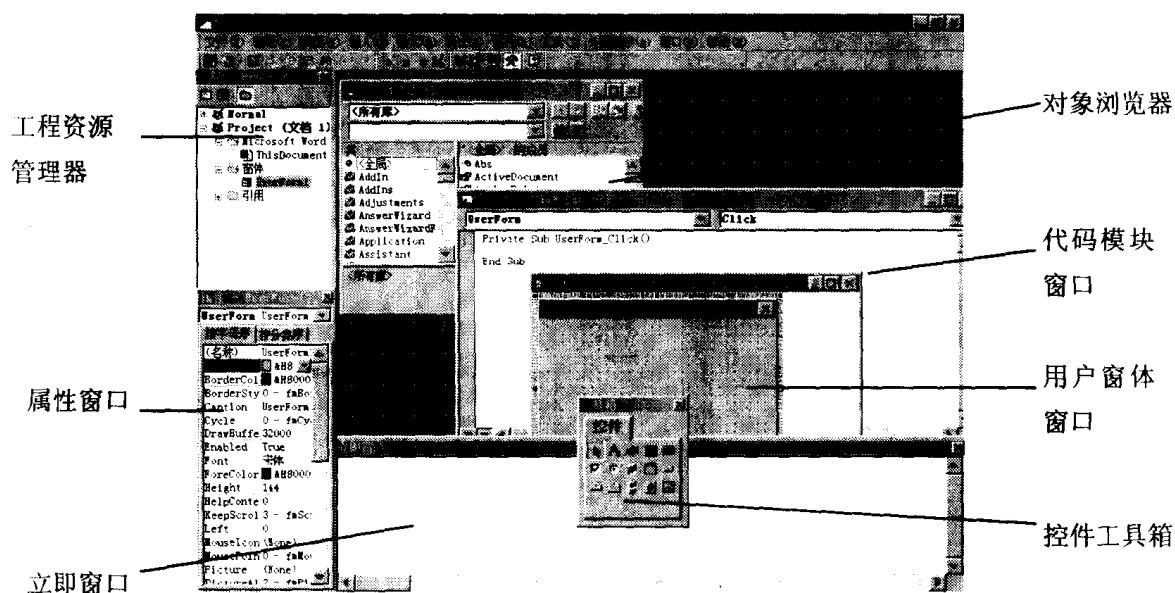


图 1-4 编辑器中主要的工具窗口

(4) 代码模块窗口:是用户编写自己程序代码的区域。VBA 提供三种标准的代码模块:标准、类和窗体。每种模块都有特定的用途。代码模块使用户将代码组织在一起以实现常用的功能。

(5) 用户窗体窗口：该窗口包含一个用户窗体(UserForm)，像所有的可视化窗口工具一样，用户可以用它来设计自己的 VBA 程序对话框。

(6) 控件工具箱：如果用户以前接触过可视化编程方面的知识，则一定不会对它感到陌生。它的使用方法和其它的可视化编程工具类似，只要将所需的 ActiveX 控件拖到用户窗体上，就可以对它进行编程来实现用户的功能。顺便提及一点，细心的读者可能会发现在 Word 2000、Excel 2000 和 PowerPoint 2000 中的工具栏里也有控件工具箱这一选项，其实它们的功能都是相似的，使用的方法也类似，只是后者是使用在相应的 Word、Excel 以及 PowerPoint 的应用程序中。

(7) 立即窗口：它的功能相当于一个代码实时监测器，用户在该窗口中键入一行 Visual Basic 代码后立即就可以看到执行结果。这对于调试程序是非常方便的。

现在我们来看看 VBA 到底有哪些功能，可以帮助用户实现怎样的功能。

用户可能对 Office 2000 中宏 (Macro) 的概念和功能还缺乏足够的认识，其实合理巧妙地使用宏可以帮助用户自动完成很多繁琐的操作，并且使用户的文档、数据等更加高效、整洁和专业化。Office 2000 提供给用户强大的宏功能，主要包括用 VBA 程序来编辑宏和使用宏录制器来录制宏。这样，通过 VBA 程序，宏实际上就完全对用户开放了。用户可以自己编写宏，也可以对录制的宏做必要的改动，以实现用户希望的功能。

Word、Excel 和 PowerPoint 都提供了创建宏的有力工具：宏录制器。Access 没有提供宏录制器，但是同样具有强大的宏功能。录制的每一个宏都保存在一个 Visual Basic 代码模块中，附在其打开的文档、工作簿或演示文稿上。下面我们先来看看怎样使用宏录制器。

比如用户想要创建一个 Excel 表格的副本，就可以使用宏录制器。

(1) 启动 Excel 2000。

(2) 在工具菜单中指向宏，然后单击录制新宏。如图 1-5 所示。

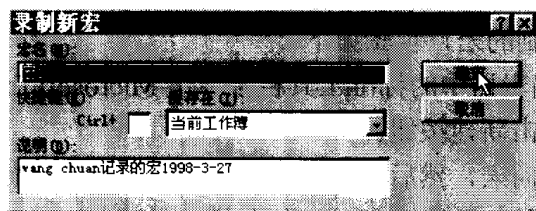


图 1-5 录制新宏

此时录制新宏对话框出现。用户指定一个宏的名称和保存的位置，则录制的新宏就保存在其中了。

单击确定按钮，此时宏录制器就开始工作了，这之后所进行的一切操作都将录制在命名的宏里面。同时还可以看到一个停止录制工具栏，单击停止按钮，则录制停止，如图 1-6 所示。用户同时可以注意到，在录制过程中，鼠标旁边有一个磁带的标志，这就表示正在进行宏录制。



图 1-6 宏录制器开始工作

- (3) 在 Excel 中输入数据, 然后单击录制工具栏中的停止按钮, 则宏录制结束。
- (4) 现在宏已经录制完毕, 按 **Alt+F11** 键启动 Visual Basic 编辑器, 在工程资源管理器中单击 Modules 旁边的十号, 便可以看到模块一。
- (5) 双击模块一, 在弹出的代码模块窗口中便可以看到录制了宏的代码。如图 1-7 所示。

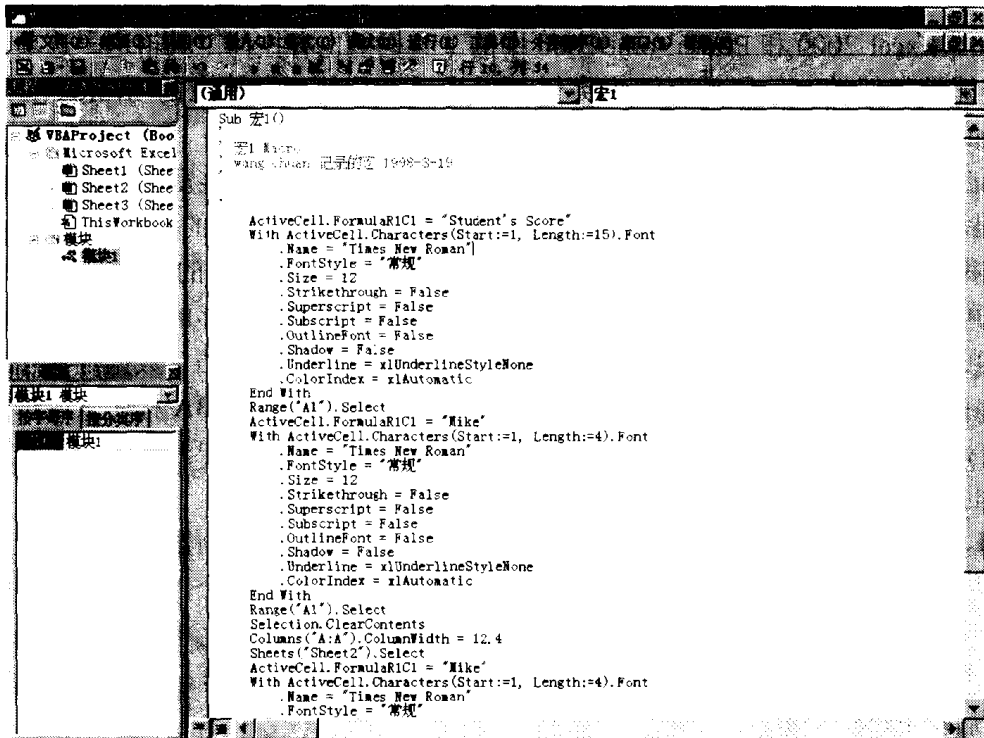


图 1-7 录制的宏代码

现在可以运行一下录制的宏, 步骤如下:

- (1) 在 Visual Basic 编辑器中的标准工具栏上单击 Microsoft Excel 按钮 , 切换到 Excel。
- (2) 在标准菜单中单击新建按钮。
- (3) 在工具菜单中指向宏, 然后单击宏, 则弹出宏对话框, 如图 1-8 所示。
- (4) 单击执行按钮, 则在新建的 Excel 表格中 Visual Basic 已经自动为用户创建了一个与先前的表格完全一样的表格。

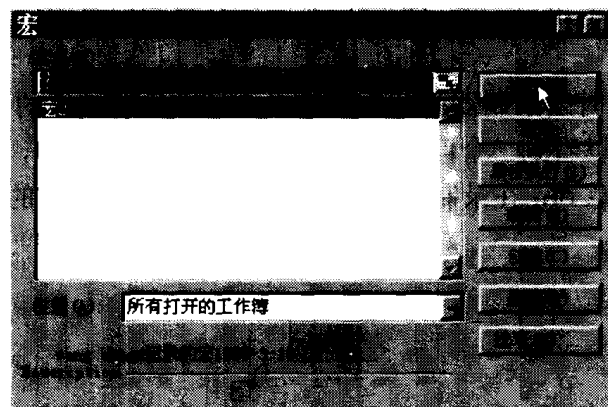


图 1-8 运行录制的宏

1.3 VBA 对象的特征

Office 2000 软件中使用的所有功能和在屏幕上所创建的所有可视内容，在 VBA 中都可由相应的对象来代表。这些对象是可编程的，因此可以开发能操纵对象属性的 VBA 程序。Office 2000 的对象集合可以划分为 Word、Excel、PowerPoint、Access 和 Outlook 对象，每个对象都包含三种成员之一：属性、方法和事件。

1.3.1 属性

可以把属性看作是重要数据的一个元素，从数值的方面而言，它描述与对象有关的性质。用户在用 VBA 构造自己的对象之前，可以提前决定对象的哪一方面应该用数字表示，然后可以设计用属性代表将遇到的数值。在 Office 2000 对象模型中，应用程序的对象已包含有属性。即使有时用户用特有的名词来表示属性，这些属于实际上代表的仍然是数字。所有属性从根本上来讲都是数值。

例如当用户若要设置属性值时，则在对象的引用后面需加上一个复合句，它是由属性名加上等号(=)以及新的属性值所组成的。例如，下面的过程可通过设置窗体中的 Caption 属性来更改 Visual Basic 窗体的标题。

```
Sub ChangeName(newTitle)
    myForm.Caption = newTitle
End Sub
```

有些属性并不能设置。每一个属性的帮助主题会指出是否可以设置此属性（读与写），或只能读取此属性（只读），或是只能写入此属性（只写）。

可以通过属性的返回值来检索对象的信息。下列的过程是使用一个消息框来显示标题，它显示在当前活动窗体的顶部。

```
Sub GetFormName()
    formName = Screen.ActiveForm.Caption
    MsgBox formName
End Sub
```

1.3.2 方法

处理对象的过程就是方法。最常用的方法是 Open，它将对象代入一个应用程序的工作空间，例如当前活动的应用程序是 Excel 2000，而现在有一个可以打开的文件名 fileName \$，则可以使用下列的指令让 Excel 2000 打开这个文件：

```
Workbooks.Open fileName $
```

被访问的 Workbooks 对象在这里是一个独特类型的对象，且相对 Visual basic 而言，它是针对 Excel 2000 的。它被称为集合并代表一组成员对象。集合对象并不等同于集合中所