

李华飏 王福水 编著

Microsoft

# Office 97/2000

## 高级使用及开发实例

—— VBA 使用指南

北京理工大学出版社

TP311.1  
L487.1

# Microsoft Office 97/2000 高级 使用及开发实例

——VBA 使用指南

李华飏 王福水 编著



北京理工大学出版社

053627

## 内 容 简 介

本书阐述了如何使用 VBA (Visual Basic for Application) 对 Microsoft Office 进行开发。介绍了 VB 的基础知识, 使得即使不懂得 VB 的读者也可以较快地入门。对 Microsoft Office 中的每一软件, 都介绍了如何使用 VBA, 包括如何使用宏, 如何创建窗体及撰写代码。对每一软件, 都介绍了如何由 VBA 来使任务自动化, 如何增强其功能, 如何使用 OLE 和 DDE 来集成 Office 的各个组件。最后介绍了在 Office 2000 中 VBA 开发的新特点。

本书适合于 Office 开发人员、Office 专业人士以及 VBA 开发人员。对于 Office 初级用户以及 VB 初学者, 跟着书中的实例学习, 也可以逐步掌握本书的内容。

## 图书在版编目 (CIP) 数据

Microsoft Office 97/2000 高级使用及开发实例: VBA 使用指南 / 李华庵、王福水编著.

—北京: 北京理工大学出版社, 1999.11

ISBN 7-81045-637-7

I.M… II.①李… ②王… III.办公室-自动化-应用软件, Office 97/2000 IV.C931.4

中国版本图书馆 CIP 数据核字(1999)第 43461 号

责任印制: 母长新 责任校对: 陈玉梅

北京理工大学出版社出版发行

(北京市海淀区白石桥路 7 号)

邮政编码 100081 电话 (010) 68912824

各地新华书店经售

国防科工委印刷厂印刷

★

787 毫米×1092 毫米 16 开本 19.75 印张 469 千字

1999 年 11 月第 1 版 1999 年 11 月第 1 次印刷

印数: 1—4000 册 定价: 32.00 元

※图书印装有误, 可随时与我社退换※

# 前言

随着 Internet 在全球的普及和应用,人们感到现代办公与自己的距离越来越近,学会使用电脑进行办公不再只是“狼来了”口号,已经成为各行各业的人们迫切要求。我们要立足现代社会、面对 21 世纪信息时代的挑战,用计算机做文档、发电子邮件、处理电子表格、制作自己的数据库、为单位制作演示已经成为各行各业的人们首先必须具备的能力。而目前全世界应用最广、影响最深、功能最强大的办公软件恐怕是微软的 Microsoft Office 了。

Office 中的 Word、Excel、Access、Outlook 以及 PowerPoint 不说都会用,至少大家接触过或听说过;而最近推出的 Office 2000 新增的 Frontpage2000、Publisher2000、PhotoDraw2000 软件大家也有所了解。但是对用于 Office 开发和编程的 Visual Basic for Application (VBA),很多人却很陌生。我想大部分读者看到这本书,首先会提出以下几个问题:

1. 这本书所说的 Visual Basic for Application (VBA)到底解决什么问题?
2. 我不想学习编程,只要能够处理日常事务就行,学会了使用 Office,为什么还要学习 Visual Basic for Application (VBA)?
3. 我以前根本不知道编程为何物,是否也能够学习 VBA 编程,这本书是否能满足?
4. 这本书与其他介绍 Office 的书籍有何不同?

在回答这些问题之前,读者可以想想,在使用 Office 的过程中,是否遇到过下面的问题或类似问题:

1. Office 功能虽强大,但它是大家的公共汽车,不是你的出租车,往往不能按照你的意愿进行工作,菜单或快捷菜单方式中有一些命令从来没有用过,而经常频繁使用的命令却得寻找半天。例如科技工作者在使用 Word 过程中由于频繁地使用上下标,他们希望把上下标变成像菜单栏上“新建”、“打开”一样的快捷命令;尤其在用 Word 97 插入标点符号时,寻找你需要的标点符号是件麻烦事。

2. Office 功能有许多简单烦琐的工作,人们不得不一遍又一遍去重复。例如在 Excel 中输入一些有规律的数据,像 1、2、3、4...这样的等差数列或像 1、4、9、16...分别是 1、2、3、4 的平方数。

3. 把 Excel 的表格插入到 Word 中,可是过了一会儿原来 Excel 的数据已经改变,又得进行一次“复制”和“粘贴”操作,能否在 Word 中调用 Excel,即让 Word 中的 Excel 表格数据保持更新?

4. 在处理单位的事务时遇到大批量的文档或数据的处理,例如处理大量的客户信息,打印大批量的邀请函(各个信函的内容相似而又有一定的区别),定期的制作最新产品的演示文稿等等,工作量非常大,你是否想过通过集成一个小型的系统来简化工作,提高效率。

.....

如果你工作中遇到这些或类似问题,那这本书无疑是为你而写的。这些问题的解决求助于 VBA (Visual Basic for Application),当然 VBA 不仅仅是解决这些问题。

## 会使用 Office，为什么还要学习 VBA？

有人也许觉得 Office 97 的功能已经足够强大了，现在又出了 Office 2000，是否还有必要使用 Visual Basic for Application 来开发 Office？的确，即使根本不用编写一句语句，Office 就可以完成用户所能遇到的各种任务。但是，人们对效率的追求是永不满足的，这就是使用 VBA 来开发 Office 的原动力。当然 Visual Basic for Application (VBA) 并不仅是解决上述问题的一些小技巧，

### 1. VBA 可以使 Office 任务自动化。

在使用 Office 的过程中，往往要遇到一些重复性的系列工作，特别是在处理大批量的文档或数据时。通过录制宏或使用 VBA 编写宏，使一系列的工作只需要一个指令便能完成，这就成倍地提高了工作效率。

### 2. VBA 可以定制 Office 满足自己的需要。

现在 Office 2000 是一个功能异常庞杂的软件包，对某一用户来说，许多命令可能一辈子也用不着，而另外一些命令可能得经常使用。有时候，功能复杂反而显得多余。

另外，Office 的外观显得异常的繁杂，许多用户面对如此繁杂的界面往往手足无措。通过 VBA 可以设置自己的界面，把自己经常用到的菜单命令和工具栏放置到显眼的地方，隐藏一些不常用的菜单或工具栏，并且，可以把自己所编写的宏设置为菜单命令或设置为工具栏中的命令按钮，方便了使用。并且，可以根据自己的需要来定制 Office 的功能，满足自己的需要。

### 3. VBA 可以增强 Office 的功能。

Office 的功能虽然强大，但它不可能考虑到所有情况，而且情况是千变万化的，各种情况都有特定的要求。通过 VBA 可以增强 Office 的功能，以处理各种各样的任务。例如，通过 VBA 可以设置 Word 拼写检查的词库，可以往词库中添加新的词语；可以设置自动更正，设置自己经常遇到的误拼等。

### 4. VBA 可以增强 Office 与用户的交互。

在 Office 中，可以利用 VBA 创建窗体，并在窗体中添加控件等用户接口元素。通过用户接口，应用程序可以得到用户的请求，并对其作出响应。

### 5. VBA 可以集成 Office 的功能。

Office 不是几个软件的简单组合，而成为一个有机的整体发挥巨大的作用。VBA 提供了集成 Office 功能的一种手段。

通过 OLE（对象的链接和嵌入）和 DDE（动态数据交换）技术的使用可以集成 Office 的功能。例如，利用 VBA，可以在 Word 文档中链接 Excel 数据表，调用 Excel 来对该数据进行处理。在 Excel 中处理该数据表，得到结果也要刷新 Word 文档中的数据表，保持了数据的更新。这时的文档确切地说应该是一个复合文档。

OLE 自动化技术则提供了一种在应用程序中控制源文档的方法。OLE 自动化的优点是，可以在单一的环境中工作，同时可以使用其他应用程序的 OLE 工具。

在 Microsoft Office 环境中，利用 VBA 开发应用程序，可以利用 OLE 自动化组合 Microsoft Office 中的多个应用程序，建立自动化应用程序。Microsoft Office 提供了大量的材料，只需

把它们拼凑起来。

### 没有学过编程，是否也能够学习 VBA 编程

既然 VBA 编程对使用 Office 这样有帮助，而且还可以编辑应用程序，那么是否与其它的语言如 VC++ 和 Delphi 等一样难学，VBA 编程与它们有什么区别？很多人一提编程就感到畏缩，因为那些并不是非职业人士所能及的。本书并不要求你是一个编程人员，只不过假定你对 Office 套件的功能有一定的了解，也就是你会使用 Office 套件。

其实很多人容易把 VBA 与 VB 混淆，二者有许多相似之处，它们的语言结构实际上难以区别，如果你会使用 VB，那么你也会使用 VBA，但不能说 VBA 就是 VB，二者是有一定区别的。VBA 是 Visual Basic 的派生体，是针对应用程序进行编程，是为了在应用程序中使用方便而有针对性地进行优化和设置了 Visual Basic。像 Word、Excel 等应用程序本身已经是成熟的软件，Access 更是数据库编程工具，所以 VBA 的主要任务是组织或者说是集合应用程序的功能。VBA 和 Visual Basic 的重要区别是：VBA 程序必须在应用程序的环境下运行；而 Visual Basic 能够直接用来在 Windows95、98、2000 以及 Windows NT 上运行应用程序。

本书阐述了如何使用 VBA 对 Microsoft Office 进行开发，同时对于一些大家比较陌生但又很实用的 Office 的高级使用技巧也做了一些介绍。介绍了 VB 的基础知识，使得即使不懂得 VB 的读者也可以较快地入门。对 Microsoft Office 中的每一软件，都介绍了如何使用 VBA，包括如何使用宏、如何创建窗体及撰写代码。对每一软件，都介绍了如何由 VBA 来使任务自动化，如何增强其功能，如何使用 OLE 和 DEE 来集成 Office 的各个组件，以实现办公的自动化。

另外有一些计算机应用人员想学编程，但是编程决非一天两天的事情，需要长期实践，然而，在使用 VBA 过程中我们能够学会编程思想和思路，这对于走向开发也是大有益处的。学任何东西关键是坚持，而要坚持下去光凭毅力是不够的，还需要建立浓厚的兴趣，编程也是这样。在 VBA 编程过程中，我们能够对编程的意义具体化，看到计算机能够按照自己的要求自动执行，一种满足感和征服感油然而生，自然对学计算机的兴趣大增。

本书是为那些想提高工作效率的 Office 用户编写的，目的是让那些想偷懒的人们少做或不做那些经常性的重复工作，以及给自己提供一个较为适合的常规和编辑选项。希望能够通过我们的一些例子起到抛砖引玉的作用，使大家不再重复那些无谓的工作，提高工作效率。

本书最后介绍了在 Office 2000 中 VBA 开发的新特点。

### 本书特色

本书与其他介绍 Office 使用或 Office 编程的书籍有何不同？其实这本书介于 Office 使用和 VB 专业编程之间，既有编程语句，又介绍了 Office 的功能，这些语句的含义很清楚而易懂，因为他们代表了 Office 的功能。

现在有关 Office 使用的书籍很多，多偏重于简单的 Office 入门介绍，而介绍高级使用的书籍涉及的又过于理论，与学习 VB 或 VC 编程的书籍没有什么区别；读者看了一时半天不易看懂，见效慢。而本书是理论联系实际，在介绍了一些基本概念以后，还有更多的应用举例，有些实例更是可以直接拿来使用，也许就可以解决你平时使用 Office 过程中碰到的问题。

本书适合于 Office 专业人士、Office 开发人员以及 VBA 开发人员。对于 Office 初级用户以及 VB 初学者，由于本书提供了大量的实例，也可以跟着实例学习，逐步掌握本书的内容。对于经常使用 Office 的应用人员既可以减轻工作量，又可以在此基础上学习编程，可谓一举多得。

在本书的编写过程中，北京科技大学的唐洪福、段瑞飞、谢长安、许廷军、吴德明以及北京航空航天大学褚洪峰等进行了部分的工作，并提供了大量的帮助，在此表示致谢。另外，特别感谢北京科技大学郭献忠博士的支持。

由于作者水平有限，错误和考虑不到之处在所难免，欢迎读者和同行批评指正。

编 者

1999 年 10 月 1 日

# 目 录

## 第 1 章 Visual Basic 快速入门

|                                                        |    |
|--------------------------------------------------------|----|
| 1.1 Visual Basic 集成式开发环境 .....                         | 1  |
| 1.2 对象、属性和方法 .....                                     | 3  |
| 1.2.1 对象 .....                                         | 3  |
| 1.2.2 面向对象的程序设计 .....                                  | 4  |
| 1.2.3 属性 .....                                         | 4  |
| 1.2.4 几个常用的属性 .....                                    | 4  |
| 1.2.5 方法 .....                                         | 5  |
| 1.2.6 实例: 滚动图片 .....                                   | 5  |
| 1.3 常用 ActiveX 控件 .....                                | 6  |
| 1.3.1 标签 (Label) 控件 .....                              | 7  |
| 1.3.2 命令按钮 (CommandButton) 控件 .....                    | 7  |
| 1.3.3 实例: 简单计算器 (Calculator) .....                     | 7  |
| 1.3.4 文本框控件 (TextBox) .....                            | 9  |
| 1.3.5 实例: 文本编辑器 (Text Editor) 程序 .....                 | 9  |
| 1.3.6 复选框 (CheckBox) 控件 .....                          | 11 |
| 1.3.7 选项按钮 (OptionButton) 控件 .....                     | 11 |
| 1.3.8 框架 (Frame) 控件 .....                              | 11 |
| 1.3.9 列表框 (ListBox) 控件 .....                           | 11 |
| 1.3.10 组合框 (ComboBox) 控件 .....                         | 12 |
| 1.3.11 滚动条控件和滑标控件 .....                                | 12 |
| 1.3.12 公共对话框 (CommonDialog) 控<br>件 .....               | 13 |
| 1.3.13 文件控件 .....                                      | 16 |
| 1.3.14 图像框 (PictureBox) 控件和图形<br>框 (ImageBox) 控件 ..... | 16 |
| 1.3.15 实例: 利用文件控件预览图形 .....                            | 16 |
| 1.3.16 时钟 (Timer) 控件 .....                             | 18 |
| 1.4 事件驱动的程序设计 .....                                    | 18 |
| 1.5 窗体的设计 .....                                        | 20 |
| 1.5.1 Windows 应用程序的风格 .....                            | 20 |
| 1.5.2 对窗体的操作 .....                                     | 21 |
| 1.5.3 菜单的设计 .....                                      | 22 |

|                           |    |
|---------------------------|----|
| 1.6 Visual Basic 语言 ..... | 22 |
| 1.6.1 变量、常量和数组 .....      | 22 |
| 1.6.2 集合 .....            | 25 |
| 1.6.3 过程 .....            | 26 |
| 1.6.4 基本语句 .....          | 26 |

## 第 2 章 Microsoft Office 开发简述

|                                          |    |
|------------------------------------------|----|
| 2.1 Microsoft Office 的组成 .....           | 32 |
| 2.1.1 Word .....                         | 32 |
| 2.1.2 Excel .....                        | 34 |
| 2.1.3 Access .....                       | 35 |
| 2.1.4 PowerPoint .....                   | 36 |
| 2.1.5 Outlook .....                      | 37 |
| 2.2 Microsoft Office 开发特点 .....          | 37 |
| 2.2.1 发展过程 .....                         | 37 |
| 2.2.2 Visual Basic 编辑器 .....             | 38 |
| 2.2.3 Visual Basic for Application ..... | 39 |

## 第 3 章 Microsoft Word 开发指南

|                                    |    |
|------------------------------------|----|
| 3.1 宏的使用 .....                     | 42 |
| 3.1.1 宏的录制 .....                   | 42 |
| 3.1.2 宏的编辑 .....                   | 44 |
| 3.1.3 宏的运行 .....                   | 45 |
| 3.1.4 宏的管理 .....                   | 45 |
| 3.1.5 为宏指定自定义工具栏和菜单栏 .....         | 46 |
| 3.2 Word 任务的 Visual Basic 实现 ..... | 49 |
| 3.2.1 Word 对象模型 .....              | 49 |
| 3.2.2 文档操作 .....                   | 50 |
| 3.2.3 文字处理 .....                   | 52 |
| 3.2.4 处理表格 .....                   | 57 |
| 3.3 Word 的定制 .....                 | 60 |
| 3.3.1 显示和修改 Word 对话框 .....         | 60 |
| 3.3.2 修改 Word 命令 .....             | 62 |
| 3.3.3 自动宏 .....                    | 63 |



|                                   |     |                                     |     |
|-----------------------------------|-----|-------------------------------------|-----|
| 3.3.4 预定义书签 .....                 | 63  | 4.3.7 Range 对象 .....                | 176 |
| 3.3.5 管理菜单栏和工具栏 .....             | 63  | 4.4 Excel 中循环语句介绍 .....             | 183 |
| 3.4 Word 工具的增强 .....              | 67  | 4.5 创建和使用用户窗体 .....                 | 185 |
| 3.4.1 使用拼写建议 .....                | 67  | 4.5.1 创建用户窗体 .....                  | 186 |
| 3.4.2 设置语言和词典 .....               | 68  | 4.5.2 添加窗体的控件 .....                 | 186 |
| 3.4.3 增强自动更正工具 .....              | 69  | 4.5.3 初始化赋值 .....                   | 187 |
| 3.4.4 使用和开发 Office 助手 .....       | 70  | 4.5.4 添加模块 .....                    | 187 |
| 3.5 Word 的集成 .....                | 73  | 4.5.5 调用窗体 .....                    | 187 |
| 3.5.1 OLE (对象的链接和嵌入) .....        | 73  | 4.5.6 窗体使用实例: 快速登记系统的建<br>立 .....   | 194 |
| 3.5.2 OLE 自动化 .....               | 76  | 4.6 Excel 开发实例: 学生成绩管理系统 .....      | 194 |
| 3.5.3 DDE (动态数据交换) .....          | 77  | 实例 1: 自动输入学号 .....                  | 195 |
| 3.6 Microsoft Word 开发实例 .....     | 80  | 实例 2: 对分数级别判定 .....                 | 196 |
| 实例 1: 文件格式转换工具 .....              | 80  | 实例 3: 两门功课总分统计 .....                | 200 |
| 实例 2: 字符统计程序 .....                | 87  | 实例 4: 按成绩进行名次排列 .....               | 202 |
| 实例 3: 同义词查询程序 .....               | 90  | 实例 5: 筛选功能自动化 .....                 | 202 |
| 实例 4: 作家助手 .....                  | 93  | 实例 6: 单科成绩统计 .....                  | 204 |
| 实例 5: 计算机考核模板 .....               | 101 |                                     |     |
| 实例 6: VCD 教学软件开发程序 .....          | 106 |                                     |     |
| 实例 7: 售后服务系统 .....                | 112 |                                     |     |
| <b>第 4 章 Microsoft Excel 开发指南</b> |     |                                     |     |
| 4.1 Excel 宏介绍 .....               | 119 | <b>第 5 章 用 VBA 对 Access 编程</b>      |     |
| 4.1.1 录制宏 .....                   | 120 | 5.1 Access 的基本使用 .....              | 213 |
| 4.1.2 查看宏代码 .....                 | 121 | 5.1.1 数据库的基本概念 .....                | 213 |
| 4.1.3 新建和编辑宏 .....                | 122 | 5.1.2 如何设计数据库 .....                 | 213 |
| 4.1.4 运行宏 .....                   | 124 | 5.1.3 利用向导建立数据库 .....               | 217 |
| 4.2 Excel 网络功能 .....              | 135 | 5.2 Access 对象模型 .....               | 223 |
| 4.2.1 数据跟踪系统 .....                | 135 | 5.2.1 Application Object 对象模型 ..... | 223 |
| 4.2.2 用 Web 发布数据 .....            | 139 | 5.2.2 Application 对象的方法及属性 .....    | 227 |
| 4.2.3 创建超级链接 .....                | 145 | 5.2.3 Docmd 对象的方法 .....             | 227 |
| 4.2.4 从 Web 上获取数据 .....           | 149 | 5.2.4 Data Access Object 模型 .....   | 231 |
| 4.2.5 开发实例: 数据查询 .....            | 151 | 5.3 使用集合 .....                      | 231 |
| 4.3 Excel 对象及其属性和方法 .....         | 157 | 5.4 使用 Access 中 VBA .....           | 232 |
| 4.3.1 Excel 对象模型 .....            | 157 | 5.4.1 创建用户界面 .....                  | 232 |
| 4.3.2 Application 对象 .....        | 159 | 5.4.2 比较 VBA 和 Access 宏 .....       | 234 |
| 4.3.3 Workbooks 集合 .....          | 167 | 5.4.3 模块及其工作方式 .....                | 236 |
| 4.3.4 Workbook 对象 .....           | 171 | 5.4.4 Visual Basic 过程 .....         | 237 |
| 4.3.5 Worksheets 集合 .....         | 174 | 5.5 控制 Application Object 行为 .....  | 239 |
| 4.3.6 Worksheet 对象 .....          | 174 | 5.5.1 打开窗体和报表 .....                 | 240 |
|                                   |     | 5.5.2 命令栏概述 .....                   | 243 |
|                                   |     | 5.6 使用 Application Objects .....    | 249 |
|                                   |     | 5.6.1 设置窗体、报表和控件属性 .....            | 249 |

|                                    |     |
|------------------------------------|-----|
| 5.6.2 数据输入的帮助 .....                | 250 |
| 5.7 使用 Data Access Object 模型 ..... | 252 |
| 5.7.1 使用记录集 .....                  | 253 |
| 5.7.2 使用查询 .....                   | 254 |
| 5.8 实例: 数据库.....                   | 255 |
| 5.8.1 切换板窗体的改进 .....               | 255 |
| 5.8.2 创建报表选择系统 .....               | 257 |
| 5.8.3 控制引入/引出操作 .....              | 260 |
| 5.8.4 创建动态选择系统 .....               | 262 |
| 5.8.5 在窗体上增加记录 .....               | 266 |
| 5.8.6 在 Access 中使用 OLE .....       | 268 |
| 5.8.7 窗体中的分页 .....                 | 270 |

## 第 6 章 Microsoft PowerPoint 开发指南

|                           |     |
|---------------------------|-----|
| 6.1 PowerPoint 对象模型 ..... | 273 |
| 6.2 实例: 产品演示模板.....       | 274 |

## 第 7 章 Microsoft Outlook 开发指南

|                              |     |
|------------------------------|-----|
| 7.1 Outlook 对象模型 .....       | 291 |
| 7.2 如何使用 Outlook 对象 .....    | 292 |
| 7.3 如何设计 Outlook 窗体 .....    | 293 |
| 7.4 Outlook 事件 .....         | 295 |
| 7.5 Outlook 开发实例: 网上表决 ..... | 295 |

## 第 8 章 Microsoft Office 2000 及其开发的新特点

|                               |     |
|-------------------------------|-----|
| 8.1 Office 2000 简介 .....      | 297 |
| 8.1.1 具有 Web 特性的协作与信息共享 ..... | 297 |
| 8.1.2 丰富的分析工具 .....           | 297 |
| 8.1.3 易于使用和管理 .....           | 300 |
| 8.2 Office 2000 开发新特点 .....   | 301 |

|            |     |
|------------|-----|
| 参考书目 ..... | 305 |
|------------|-----|

# 第1章 Visual Basic 快速入门

本章主要内容:

Visual Basic 集成式开发环境

对象、属性和方法

常用 ActiveX 控件

事件驱动的程序设计

窗体的设计

Visual Basic 语言

Visual Basic 不仅仅是一门计算机语言,而且是一个集应用程序开发、测试、查错功能于一体的集成式开发环境。Visual Basic 从产生以来,经过发展,已成为一门强大的开发工具,除了开发传统应用程序外,还可以完成以下工作:

利用对象的链接与嵌入(OLE)技术进行对象编程,数据库系统编程,语音与图像的应用及处理,利用 ActiveX 和 VBScript 开发 Web 应用程序等。而内置在 Microsoft Office 中的 Visual Basic 编辑器,则极大地增强了 Office 的功能,提高了工作效率。

必须提请注意的是,讲述本章的依据是独立的 Visual Basic for Windows,而且本章的实例都是在该环境下调试通过,而不是在 Office 环境下的 Visual Basic for Application。因为 Visual Basic for Application 是 Visual Basic 的派生体;而且学习本章的目的是尽快地掌握 Visual Basic 这门语言,以及了解 Visual Basic 这种开发环境。

## 1.1 Visual Basic 集成式开发环境

启动 Visual Basic,并选择想要开发的工程(Project,也称项目)类型后,可以看到 Visual Basic 集成式开发环境(IDE),如图 1-1。

Visual Basic 的开发环境包括以下几个部分:

菜单栏

工具栏

工程资源管理器(Project Explorer)

工具箱

属性窗口

窗体设计器

窗体布局窗口

## 立即(Immediate)窗口

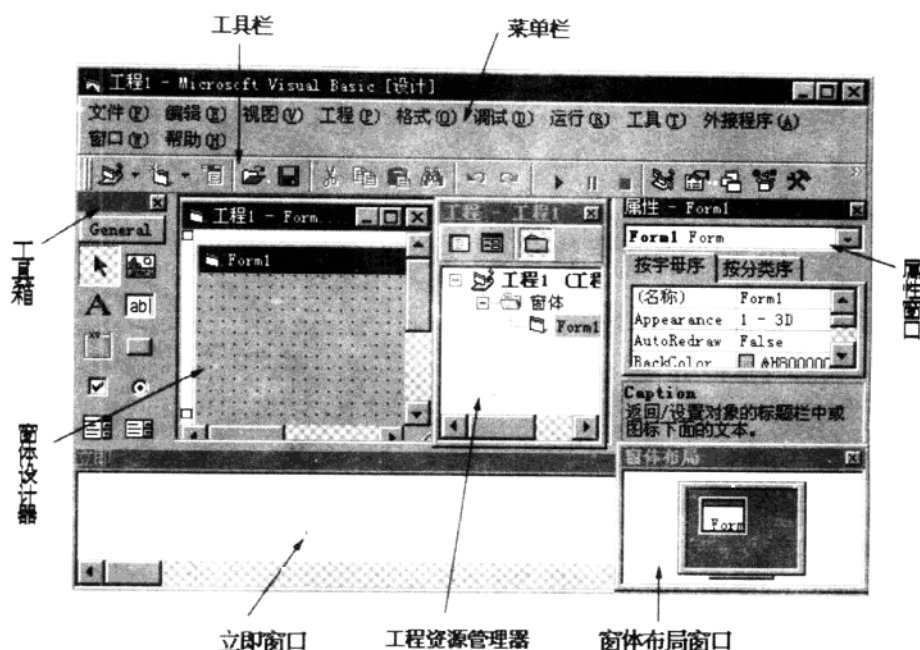


图 1-1 Visual Basic 集成式开发环境

## 1. 菜单栏

菜单栏包括使用 Visual Basic 所需要的命令。其菜单有：

**文件(File)** 包含打开和保存工程、保存窗体以及生成可执行文件的命令，并显示一系列最近打开的工程。

**编辑(Edit)** 包含编辑命令，如“撤消”、“复制”、“粘贴”等命令和许多格式化、编辑代码的命令。

**视图(View)** 控制集成式开发环境各个组成部分(代码窗口、对象窗口和工程资源管理等)的显示和隐藏。

**工程(Project)** 包含将构件加入当前工程、引用 Windows 对象和往工具箱添加控件的命令。

**格式(Format)** 包含对齐窗体中控件的命令，以及调整其大小，以保持外观的整齐。

**调试(Debug)** 包含查错命令等。

**运行(Run)** 包含启动、断点设置和终止当前程序运行的命令。

**工具(Tools)** 包含建立 ActiveX 构件和 ActiveX 控件时所要的工具，以及启动菜单编辑器的命令和配置环境的“选项”命令。

**外接程序(Add-Ins)** 包含可添加或删除的插入件。

**窗口(Window)** 包含窗口布局的命令。

帮助(Help) 包含帮助信息。

## 2. 工具栏

工具栏中放置使用次数较多的菜单命令, 使得用起来更加方便。Visual Basic 中有标准工具栏、编辑工具栏(放置编辑菜单的命令)、查错工具栏(放置调试菜单的命令)、窗体编辑工具栏(放置格式菜单的命令)。

## 3. 工程资源管理器(Project Explorer)

工程资源管理器(Project Explorer)显示工程的组成结构。

## 4. 工具箱

工具箱中放置控件的图标。创建应用程序的接口时, 直接从工具箱中调用, 再放置到窗体中。

在缺省情况下, 工具箱中有近 20 个控件图标。可以往其中添加更多的控件图标。

## 5. 属性窗口

属性窗口放置对窗体和各种控件的属性值。通过属性窗口, 可以设置控件的属性。

## 6. 窗体设计器(Form Designer)

窗体设计器(Form Designer)用来设计和编辑应用程序的用户接口。对于每个窗体, 窗体设计器(Form Designer)显示两个窗体:

对象窗口(用户接口元素, 包括窗体和控件)。

代码窗口(对象的代码)。

## 7. 窗体布局窗口

窗体布局窗口用于确定窗体在应用程序中的初始位置, 及窗体间的相对位置。

## 8. 立即(Immediate)窗口

立即(Immediate)窗口是辅助查错窗口。

# 1.2 对象、属性和方法

## 1.2.1 对象

Visual Basic 是一种面向对象(Object)的语言。将数据和处理该数据的过程(函数和子程序)打包在一起而生成的新的数据类型称之为对象。在 Visual Basic 中, 所有的窗体和控件都是对象。窗体是可用做定制应用程序界面的窗口, 或用作从用户处收集信息的对话框。控件的类型包括文本框控件(TextBox Control)、列表框控件(ListBox Control)、组合框控件(comboBox Control)、滚动条控件(ScrollBar Control)、公共对话框控件(CommonDialog Control)等等。这些控件将在后面详细介绍。

## 1.2.2 面向对象的程序设计

面向对象的设计方法把系统中的所有资源,如数据、模块以及系统都看成对象,每个对象把一组数据类型和一组过程封装在一起,使得这组过程了解对这一数据类型的处理,并在定义对象时可以规定外界在其运行上的权限。使用这一方法,设计人员可以依照自己的意图创建自己的对象,并将问题映射到该对象上。

对象是一个功能和数据独立的单元,它有其自身的属性、方法和可为别人所做的工作,互相之间只能通过对象认可的途径进行通讯来完成任务,相互之间没有预料不到的影响,也可以较为自由地为各个不同的软件所使用。这种代码共享,避免了重复的代码设计,提高了开发软件的效率。

另外,面向对象系统基于这样一个假设,认为系统和程序员共享一个共同的环境,并且所有程序都是对该环境的扩充。因此,面向对象语言对程序设计的主要影响并不在于它的语法特征,而在于它提供了一个结构,一个开发环境,据此可以装配所有的程序,并建立程序单元之间的通讯。

## 1.2.3 属性

和现实世界的事物一样, Visual Basic 中的对象有其属性(property)。若把 Visual Basic 中的对象看作名词,那么,属性便是它的形容词。属性是对象的一个特性或者该对象操作的一个方面。例如,文本框的属性包括 Backcolor、Forecolor、Enabled、Height、Width、Left、Top、Name 等属性。要更改一个对象的属性,可以修改属性的值。要设置一个属性的值,可在对象的后面带一个符号“.”、属性名、一个等号 and 新的属性值。

例如,一个窗体(Form1)的属性可以设置如下:

```
Form1.BackColor=#ff0000
```

```
Form1.BorderStyle=1
```

```
Form1.Caption="Form1"
```

```
Form1.WindowState=2 'Maximized
```

这是一个背景为红色、最大化窗口、边框可见、标题条文字为“Form1”的窗体。

## 1.2.4 几个常用的属性

Visual Basic 中的窗体和控件都有其各自的属性。不过,有一些属性适用于大多数对象:

**Name** 该属性设置控件名称,作为访问控件的属性和方法的依据。

**Appearance** 设置控件的外观。

**BackColor** 设置显示文字或绘制图形时的背景颜色。

**ForeColor** 设置前景颜色。

**Font** 设置控件文本所用字体的种类、字号和属性。

**Caption** 设置一些不接受输入的控件上显示的文本,如标签控件的文本。

**Text** 设置接受输入的控件上显示的文本,如文本框控件的文本。

**Width、Height** 设置控件的尺寸。

**Left、Top** 设置控件的左上角坐标。

**Enabled** 设置控件是否激活。缺省情况下为 True;设置为 False 时控件不可用,颜色变为

灰色。

Visible 设置控件是否可见,缺省时为可见。

### 1.2.5 方法

Visual Basic 的对象也有动作和行为,也就是说它们“知道”该怎么做。这些动作就叫做对象的方法。方法就像是动词。例如,窗体的显示或隐藏就是方法。上面的窗体 Form1 的显示可写为 Form1.show。

在大多数情况下,方法是动作,而属性是性质。使用方法会使对象发生某些事情,而使用属性则会返回对象的信息,或者引起对象的某个性质的更改。

### 1.2.6 实例:滚动图片

了解 Visual Basic 中的对象、属性和方法后,我们便可以设计一些简单的程序。例如,我们要设计一个滚动画面。便可在窗体中放一个图片,改变图片的 Left 属性值,让图片从左向右移动。具体步骤如下:

新建一个窗体,往其中添加一个图片框控件(Image Box)、一命令控件(CommandBar Control)、一时钟控件(Timer Control)。

图片框控件取名为“Image1”,将其 Picture 属性中设置一要放入图片的文件名,如“C:\1.bmp”;其 Left 属性设置为 0。

时钟控件取名为“Timer1”,其 Interval(时间间隔)属性设置为 20(毫秒),Enabled 属性设置为“False”。

在代码窗口中键入如下代码:

```
Sub Timer1_Timer()  
Static CurrentLeft  
CurrentLeft=CurrentLeft+200  
If CurrentLeft < 6000 then  
Image1.Left=CurrentLeft  
else  
CurrentLeft=0  
End If  
End Sub
```

系统每隔 20 毫秒激发一次时钟,即运行一次 Timer1\_Timer()子程序。在这里利用一个静态变量 CurrentLeft 来不断的改变图片框 Image1 的 Left 属性值,使它循环往复地从左向右滚动。

命令控件取名为“Commd1”,Caption 属性设为“滚动”。在代码窗口键入如下代码:

```
Sub Commd1_Click()  
Timer1.Enabled=True  
End Sub
```

由于 Timer1 的 Enabled 属性设为 False(不可用),所以由命令控件 Commd1 的 Click 事件来驱动时钟。

这时执行程序,窗体中有一静止图片,以及一“滚动”按钮。用鼠标点击按钮,该图片

便不停地从左向右滚动。

### 1.3 常用 ActiveX 控件

窗体是 Visual Basic 应用程序中的顶层对象，每个应用程序都是从窗体开始的。窗体是构成用户接口的所有控件的容器。而控件是用户操作对象的图形表示，以便向应用程序提供信息，如：按钮、列表框等。有的控件没有图形表示，如时钟(Timer)控件、Winsock 控件等。常用控件的图标都显示在工具箱中，如图 1-2 所示。



图 1-2 工具箱

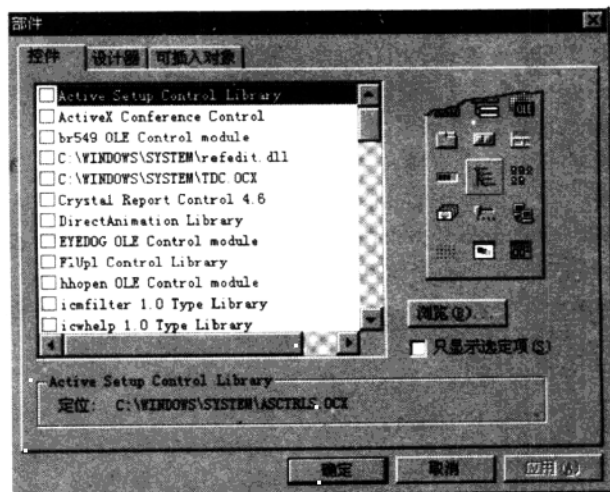


图 1-3 部件对话框

有的控件的图标没有在工具箱中，可以通过以下步骤调用：

- (1) 单击“工程”菜单的“部件”命令，弹出“部件”对话框，如图 1-3。
- (2) 从“部件”标签中选择想要的控件，单击“确定”。

这时，相应的控件图标便出现在工具箱中。

常用的控件有以下几种：

|      |               |       |              |
|------|---------------|-------|--------------|
| 标签   | Label         | 滑标    | Slider       |
| 命令按钮 | CommandButton | 驱动器列表 | DriveListBox |
| 文本框  | TextBox       | 目录列表  | DirListBox   |
| 选项按钮 | OptionButton  | 文件列表  | FileListBox  |
| 复选框  | CheckBox      | 公用对话框 | CommonDialog |
| 框架   | Frame         | 图形框   | ImageBox     |
| 列表框  | Listbox       | 图像框   | PictureBox   |
| 组合框  | ComboBox      | 时钟    | Timer        |
| 滚动条  | ScrollBar     |       |              |



### 1.3.1 标签(Label)控件

标签控件主要用于在窗体中显示提示信息,常与 Text 控件一起使用。

标签控件的主要属性是 Caption 属性,它表示标签的内容。例:Label1.caption="E&xit",控件在窗体中显示 Exit;&后面的字母是快捷键字母,即当按下 Alt+X 时,Label1 的 click 事件将发生。这种快捷键的定义方法对于菜单和许多其它控件同样适用。

### 1.3.2 命令按钮(CommandButton)控件

命令按钮控件用于接受用户的命令。

### 1.3.3 实例:简单计算器(Calculator)

利用标签控件和命令按钮控件,以及一些简单的代码,便可以编写一些小程序了。

例如,编写一个简单计算器,其外观如图 1-4 所示。单击“Star”按钮便可以开始计算。通过数字按钮输入运算的第一个数,再单击将要进行的运算按钮,然后输入第二个数。这时,单击“=”按钮便可以得到结果。

具体的步骤如下:

新建一窗体,窗体命名为 Cal,Caption 属性为“Calculator”。在窗体中添加一个标签(Label)控件,以及数个按钮。

标签(Label)控件命名为 Label1,Caption 属性设为“开始计算,请按下 Start”,将其 BorderStyle 属性设为 1-Fixed Single,使其出现三维外观。

用 10 个命令按钮控件来分别代表 0~9 十个数字,依次命名为 C0,C1,...C9,Caption 属性依次为“0”,“1”,...“9”。

用四个按钮分别表示加减乘除,依次命名为 Cjia,Cjian,Cchen,Cchu,Caption 属性分别为“+”、“-”、“\*”、“/”。

另外用一个按钮表示开始计算,命名为 CStart,Caption 属性为“Start”。用一个按钮来得出结果,命名为 Cresult,Caption 属性设为“=”。

最后的界面如图 1-4。

程序的代码如下:

1. 声明变量。

声明三个变量,代码如下:

```
Dim a1, a2, result As Double
```

```
Dim Cal As Integer
```

2. 使用 Start 按钮初始化变量。

```
Private Sub Cstart_Click()
```

```
Label1.Caption = ""
```

```
a1 = 0
```

```
a2 = 0
```

```
result = 0
```

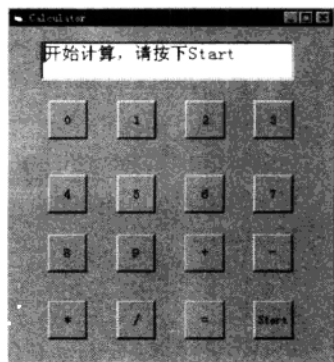


图 1-4 计算器程序的外观