中文版 Office 2000 VBA 一册通

王　川　　阮兴平
于平剑　　何建波　　编著

人 民 邮 电 出 版 社

需要全本请访问 www.ertongbook.com

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。

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够编出合格的程序，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、曾颍峰、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

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。

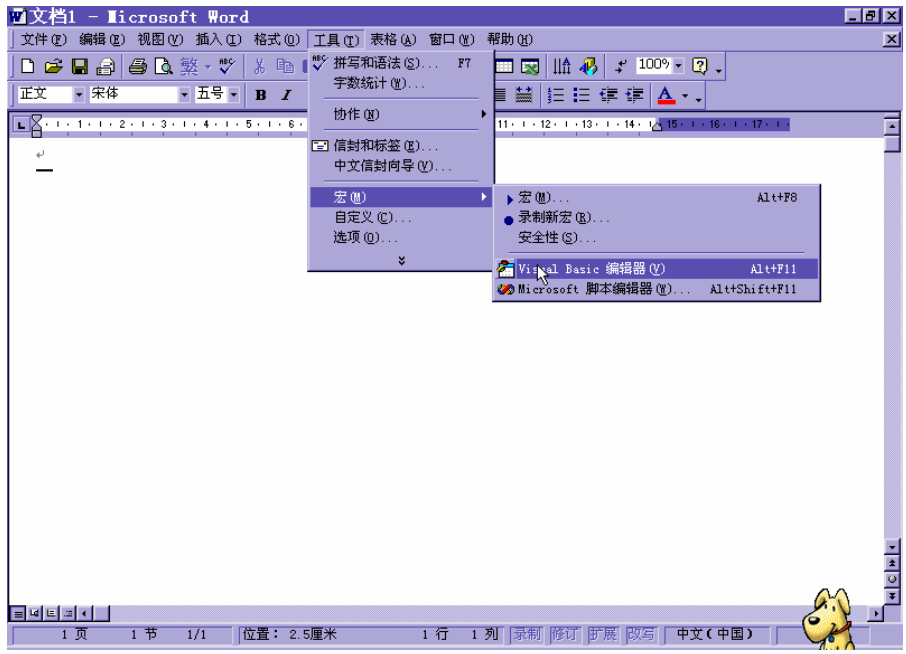


图 1-1　打开 Visual Basic 编辑器

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。

需要全本请访问 www.ertongbook.com

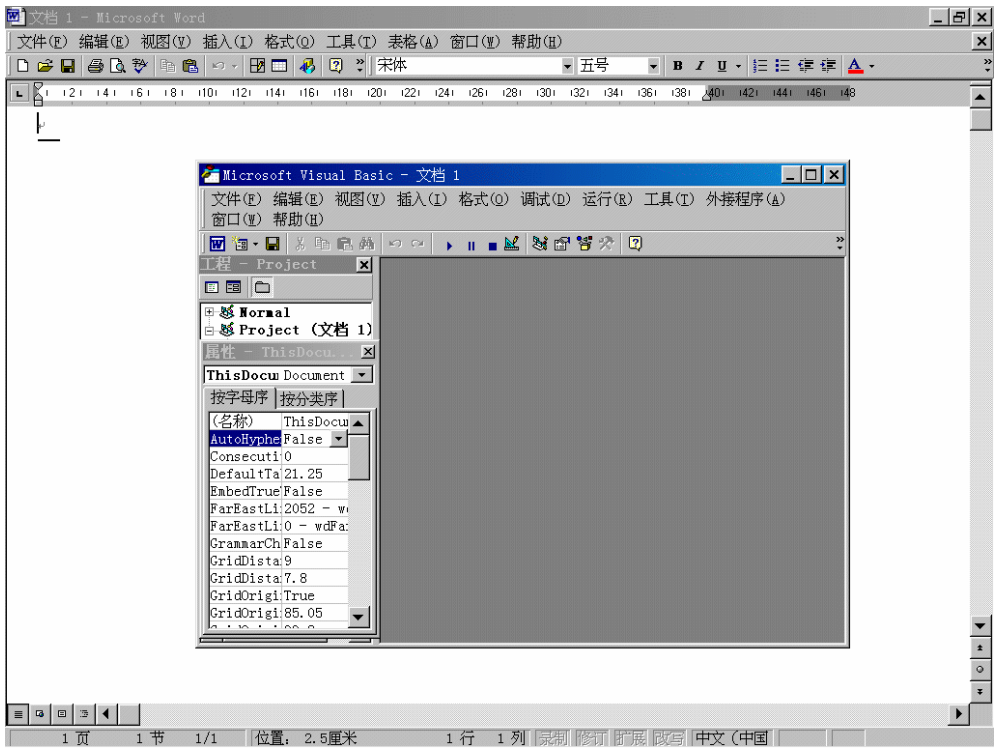


图 1-2　Visual Basic 编辑器界面

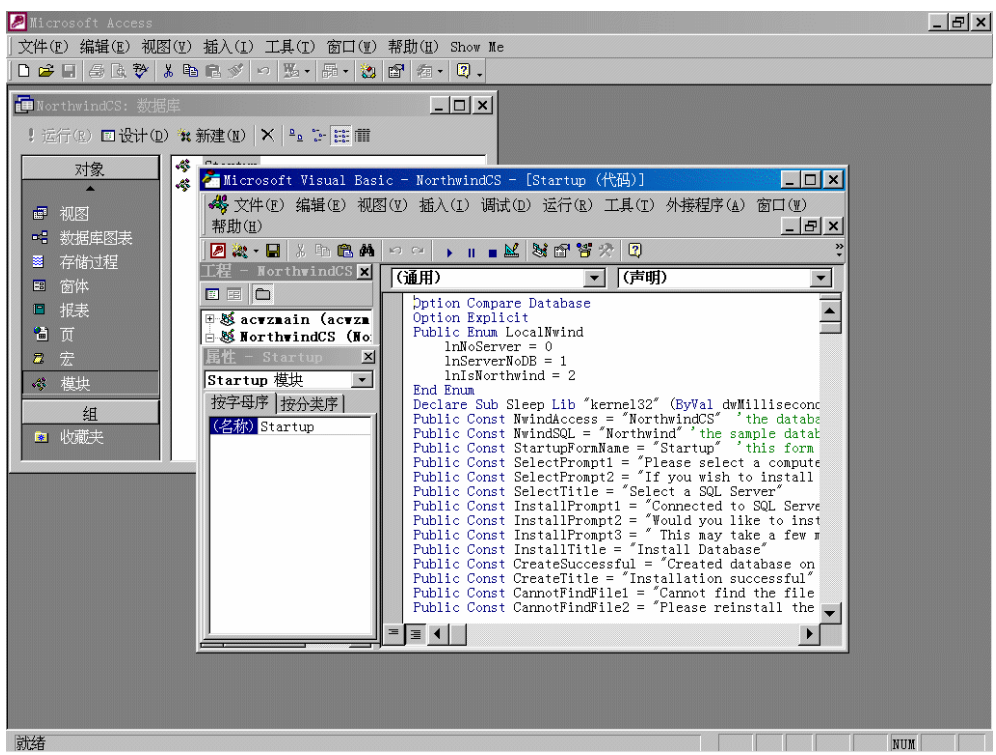


图 1-3　Access 2000 的编辑窗体

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。

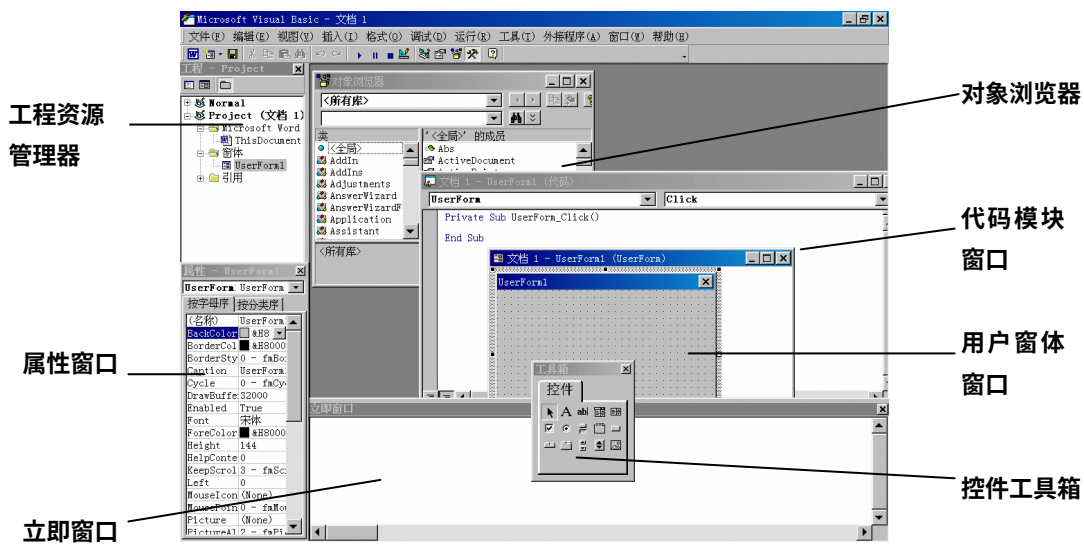


图 1-4 编辑器中主要的工具窗口

(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。它的使用方法也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＋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。

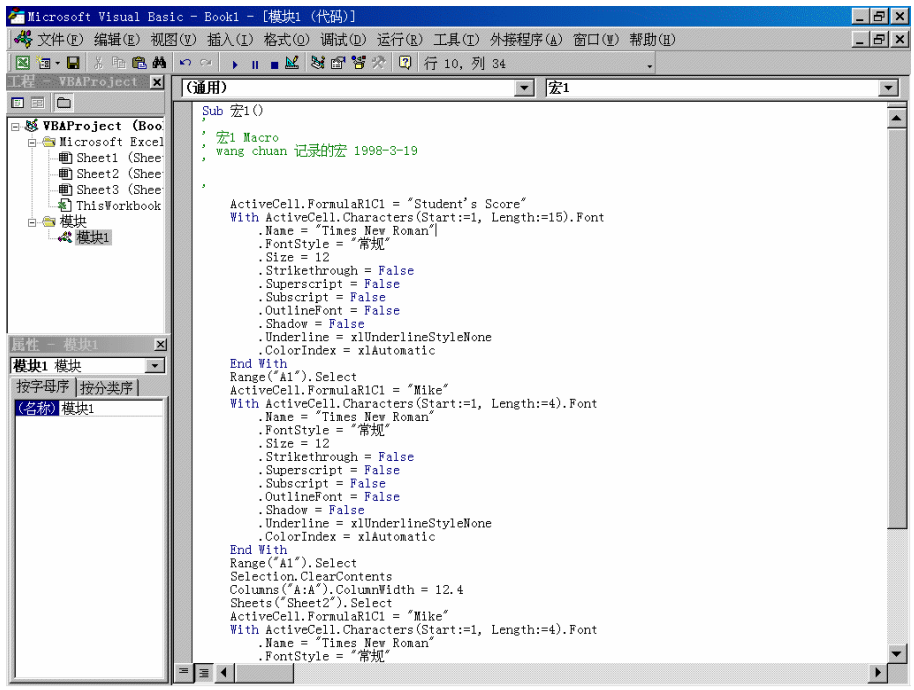


图 1-7 录制的宏代码

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 Wookbooks 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有

的对象；相反它代表的是集合的成员结合在一起的对象。它有点类似于 C++中的类，只是在这种情况下，强调的是一个整体而不是集合中的特定成员。在 VBA 编辑器解释这条指令时，文件名 fileName $是作为 Open 方法的一个参数。当文件被打开时，就作为 Workbooks 的成员。

在下面的例子中可以看到，Add 是属于 ComboBox 对象的一个方法，因为它会增加一个新项到下拉式列表框中。

下面的程序是使用 Add 方法增加一个新的工程到 ComboBox 中：

```
Sub AddEntry(newEntry as String)
    Combo1.Add newEntry
End Sub
```

1.3.3 事件

对象的另一个重要特征是事件。事件是为了帮助定义执行一个事件响应过程的术语。只有一部分 Office 2000 对象使用事件。但实际上在用户构造通常对话框时使用的所有 ActiveX 空间都使用了事件。这里有两个简单的过程说明事件是怎样在用户的程序里起作用的，下面以 Document 对象为例。

下例显示了 Normal 工程中的一个 New 事件过程。在新建一篇基于 Normal 模板的文档时，运行此过程：

```
Private Sub Document_New()
    MsgBox "New document was created"
End Sub
```

下例显示了文档工程中的一个 Close 事件过程。该过程只在关闭文档时运行：

```
Private Sub Document_Close()
    MsgBox "Closing the document"
End Sub
```

每个对象都有它的生存期，在其中它要经历某些事件，例如被打开，被关闭等等。这些事件在所响应的对象库中都有特定的术语相对应；这些术语在 VBA 的对象库中都已经规定了，用户在需要的时候可以通过各种工具来查询。

关于对象的属性、方法和事件的详细内容，请用户参阅第二章的有关内容。

1.4 VBA 工程的结构

VBA 在构造一个工程时调用一个程序。工程对于 VBA 而言是一个整体的概念，它包括 Word 文档、Excel 工作表格、PowerPoint 演示和其它的 Office 应用程序所构成的所有文件，

需要全本请访问 www.ertongbook.com

它们都可以称之为一个工程。Normal 模板包含一项应用工程中所有文档都分享的功能。特定模板包含对某一特殊类型文档特殊的功能。

工程中的两个重要概念是过程和模块。这三者间的关系就像在我们的工作中，当用户面对一项工程时，就将一个工程分成几个模块，将这些模块交给不同的人来完成一样。这些模块中又包含了多个过程。

过程是指令的集合。它的目的就是完成最终模块的至少一项任务。在如下的一段 VBA 代码中，用户可以更清楚地知道什么是一个过程：

```
Sub DemoBox()      '该过程声明一个字符串变量，
                   '指定它的值为 Claudia，然后
                   '显示一个连接的消息。
    Dim myVar As String
    myVar = "John"
    MsgBox Prompt:="Hello " & myVar, _
        Title:="Greeting Box", _
        Buttons:=vbExclamation
End Sub
```

在这一段代码中，用户可以看到一个完整的过程，它是从 Sub 开始，以 End Sub 结束的。这是一个简单的过程，一个过程还可以包含对其它过程的调用，这是它完成其它代码所提供的功能的方法。这些过程调用常包含参数，这些参数可使过程得到特别针对用户工作的数据提示。接收过程就利用这些参数来完成一些功能。一个简单的例子就是让过程画一个长方形，过程本身并不含长方形的确切位置，这些坐标由调用过程提供，所以画长方形的过程中就包含子过程，这个子过程含有长方形确切位置的坐标信息。

模块是过程的集合，一组相互之间传递控件的 VBA 过程就是模块。比如说，当我们在读一本书的时候，第一章里写到：“详细说明请参考第十章”，而在第十章里有：“在第五章里也有类似的内容”。同一模块中的另一过程可以通过名字调用这一过程来传递控件。如果一个过程需要它所调用过程的数据，则这些数据将会从括号中的区域传递过来。

VBA 并不像其它 Visual Basic 编程环境。VBA 可以管理三层工程，或超过三层的工程。层之间的属性关系如下：

(1) Normal 工程：该工程包含在任何时刻应用程序的所有文档均能得到的模块和过程。对于一个应用程序而言，只有一个 Normal 工程，因为应用程序只有一个 Normal 模板。

(2) 模板工程：该工程包含属于应用程序产生的特殊类文档的代码，在任何时刻，在应用程序中有多少活动的模板就有多少打开的模板工程。所以，如果用户在 Excel 中打开了三个工作表，其中两个使用同一种模板而第三个使用第二种模板，那么在 Excel 的 VBA 窗口中就有两个活动的模板工程。

(3) 文档工程：该工程包含专属于一个打开文档的代码和特征数据。VBA 自动产生与打开文档同样多的文档工程，即使文档工程中不包含 VBA 源代码。

VBA 并不具备完整的面向对象语言的特点，但是从另一个角度来说，它处理模板的方式

与面向对象的继承概念有相同的地方。实际上 Office 2000 应用程序的 Normal 模板定义了原始的“类”，即所安装的系统上的应用程序的功能。所以，和独立模板一同产生的文档从某种程度上说是所谓多级继承的结果。它继承了两类文档的功能，被两个模板定义。文档可以用自己的功能来延伸这种继承的功能。

1.5 Windows 的组件模型

Office 2000 采用的是和 Windows 类似的以组件模型为基础的运行机制。例如，Office 2000 的网络功能是由 Internet Explorer 提供的。Word 2000 和 Excel 2000 中的拼写检查器是由名为 Proof 提供的一种拼写检查器。Equation Editor、组织图表生成器、格式转换器，甚至是 VBA 本身都是一种自维护的 ActiveX 组件。

在 Office 2000 的应用程序中，通过用户的交互式操作所产生的主要的数据就是文档。要提醒用户注意的是文档并不是一个文件。因为在一个文档当中，由于插入了各种组件和其它文档数据，所以一个文档可能是由几个文件的数据组成的。在文档中插入的诸如图形和控件等，当用户双击它们时，很可能就会影响到所涉及到的组件。

例如，若用户在自己的 Word 2000 文档中插入一张 AutoCAD 图片，如图 1-9 所示，则在插入菜单中单击对象，便出现如图 1-10 所示的对话框。

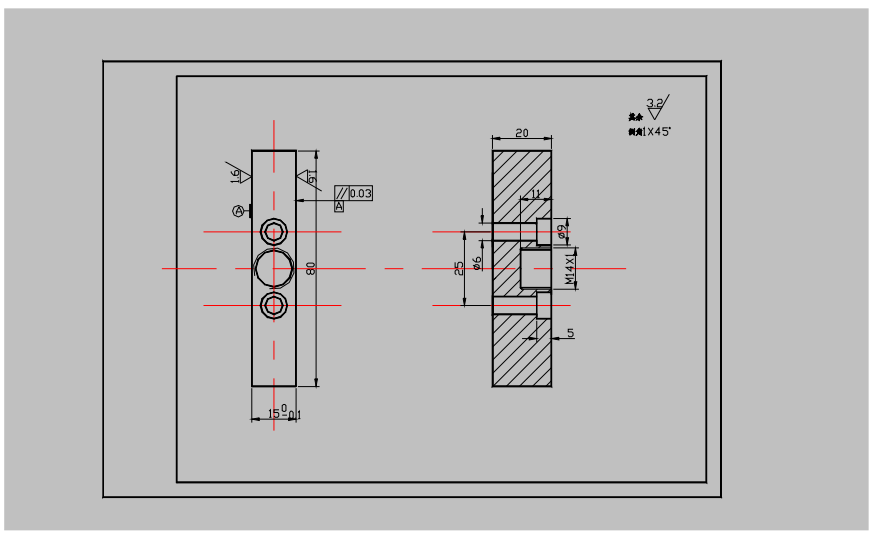


图 1-9　要插入的 AutoCAD 图片

在其中选定 AutoCAD Drawing 项，单击确定，在完成图片的插入后，该图片仍然保持着它原来的组件 AutoCAD 的特征。当用户双击该图片时，系统会自动打开 AutoCAD 图形编辑界面，该图片就在其中，等待用户的编辑。从这个例子可以看出 Office 2000 的组件模型的特点。

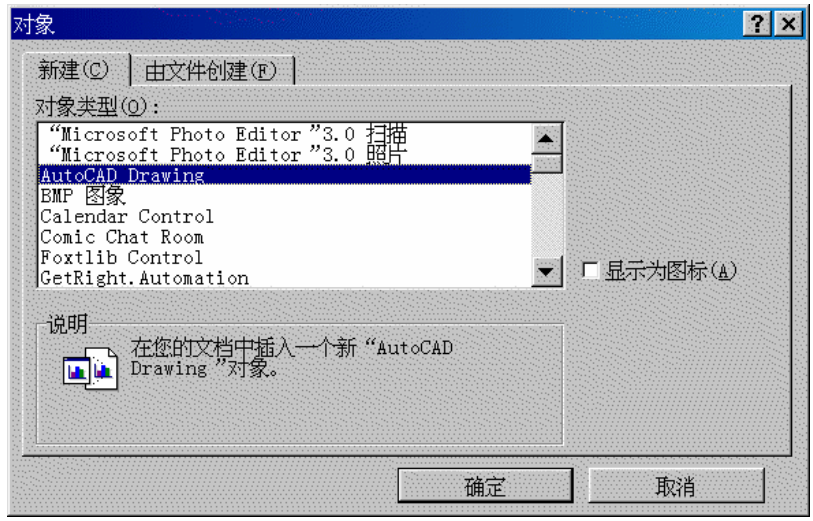


图 1-10　插入对象

在 Office 2000 **组件内部也可以实现这样的互相插入和互相调用，**Word 2000、Excel 2000、PowerPoint 2000 **和** Access 2000 **等本身就是一些典型的** Windows **组件。它们既能独立工作又可协同工作。这意味着，**VBA **程序是在一个来自独立的组件的一个或者多个对象组成的"** Office 2000 **文档"中进行编程的。这样的特点建立在对象特征的基础上，它们的核心是** Application**。当用户为** Access 2000 **编写模块时，**Application **对象代表** Access **；而在为** Excel 2000 **编写模块时，**Application **对象代表** Excel**。在这两种情况下，**Application **的成份是不同的，但是不变的是** Office 2000 **所固有的组件模型特征，这在以后的介绍中将会很明显地体现出来。**

第二章　Microsoft Office 2000 的对象模型

2.1　什么是对象模型

在任何计算机语言中，对象代表着程序的一些特点，或由程序产生的数据的某个方面。作为表示，对象用符号表示数据，但也作为那个数据的代理。对数据结构所做的改变或数据处理、存储和显示方式上的变化均作为对对象的请求，作为数据指定表示被访问。任何被用户认为是合理的事物如人、树木、街道等都可以被视为对象。对计算机而言，程序中的对象是由其基本功能所定义的。对象模型（Object Model）代表应用程序中的元素，比如工作表、单元格、图表、窗体，或是一份报告。在 Visual Basic 的代码中，在使用对象的任一方法或改变它的属性之一的值之前，都必须去识别对象。

2.1.1　原始对象

对象是 Visual Basic 这一宏伟大厦的基石，在 Visual Basic 中的每个操作都与修改对象有关。Word 的任何元素（如文档、段落、域、书签等等）都可以用 Visual Basic 中的对象来代表。

作为一个例证，VBA 不能直接在开放空间中标识 Document 对象。用户可以为之声明一个具有 Document 类型的 VBA 对象变量，让它代表某个文档，并在以后赋予那个变量。但是你不能对那个对象变量有任何操作，除非让它代表一个真正的 Document 对象。所有这些都表明，用户不能标识一个与真正文档无任何联系的 Document 对象。VBA 为了在 Document 对象和真正文档之间建立真正联系，Document 对象必须植根于一个属于正运行的 Office 2000 应用程序的原始对象中。

当用户在任何一个 Office 2000 中启动 Visual Basic 编辑器时，Visual Basic 编辑器都假设原始对象即阶层中最高阶层的对象是与那个应用程序相联系的对象库的名字。表 2-1 列出了四个主要 Office 2000 对象库的原始对象。

表2-1　　Office 2000 对象库的原始对象

对象库	原始对象
Word 2000	Word
PowerPoint 2000	PowerPoint
Access 2000	Access
Excel 2000	Excel

这些原始对象的所有成份代表了程序的所有可标识部分和这些程序产生的所有可标识的数据。

当用户在 Visual Basic 编辑器中为一个应用程序工作时，不必在每次引用对象时都指出原始对象。例如，下面就表示 Word 中光标处于其上的文档的对象：

Word.activeDocument

当用户在 Word 的 Visual Basic 编辑器工作时，Word 被认为是缺省的，所以如果用户写成：

ActiveDocument

则 Visual Basic 编辑器就认为这与 Word.ActiveDocument 是一样的。在 Word 2000 中 ActiveDocument 是对象库全局对象中的一个，这意味着它可以不必向编辑器表明就可以维持它的有效性。

那么用户在何时需要使用到原始对象名呢？

这就是当用户需要从另一个应用程序的 Visual Basic 编辑器中引用一个应用程序时。假设用户的 Excel 过程要用对象变量来引用用户的 Word 文档，过程将声明那个变量，则将使用如下的指令：

Dim objWordDoc As Word.Document

这条指令只有在用户设置 Excel 2000 的 Visual Basic 编辑器以识别 Word 2000 对象库时才有效。这条指令声明了一个新项 objWordDoc 作为 Word 中当前打开文档的唯一的引用。这种形式的变量称为对象引用。Word Document 是这种新的对象引用的对象类型。一个对象类型项用于告诉编辑器来识别这个新变量，从而赋予它 Word 文档对象的所有特点。

一旦 Excel 2000 的 Visual Basic 编辑器执行这条指令，变量 objWordDoc 就是有效的引用，但还需要指向一个真正的文档，所以就需要如下的指令：

Set objWordDoc = ActiveDocument

此时变量 objWordDoc 就指向了 Word 中的任何活动文档了，在这以后如果用户需要决定 objWordDoc 指向的文档的文件名，可以用如下的指令：

sFileName = objWordDoc.Name

这样，字符串变量 sFileName 通过 Name 属性就被赋予了文件名。

2.1.2　中央对象和全局对象

所有的 Office 2000 应用程序的中央对象都是 Application。之所以要设置一个对所有对象库都共有的对象项，并让所有子项与之紧密联系的原因是 Office 2000 Object Model 中的大部分成员是双向的，Application 与每一个子项都存在着相互联系。这些子项在 VBA 模块的任何地方都可以应用，故称之为全局对象。正是 Application 联系着这些全局对象，所以全局对象是 Application 的成份。

图 2-1 表示了中央对象和全局对象的关系。

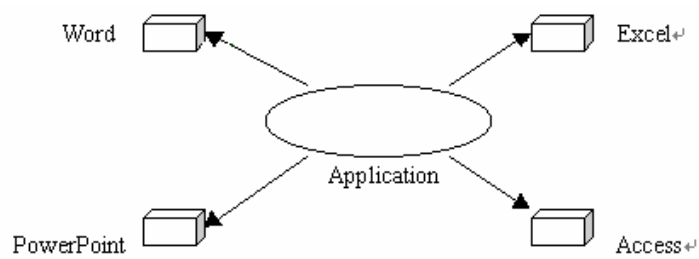


图 2-1　中央对象和全局对象的关系

对于一个全局对象来说，调用 Application 就像它是一个属性一样。这有点像 Object Model 中的一个备份，在唯一的数据和用户需要的数据没有直接关系的情况下，用户可以用 Application 对象作为中介。

全局对象不必为了被识别而需要和各自库的根项相联系，即使在它们被输入到其它的编辑器中也是这样。当用户编写的 VBA 指令均以某种方式包含一个全局对象时，即使它是缺省的或缺省该项也是这样。表 2-2 列出了 Office 2000 Object Model 库的全局对象。

表2-2　　四个主要Office 2000 Object Model 库的全局对象

对象库	全局对象	
Word 2000	ActiveDocument	KeyBoundTo
	Activewindow	LandscapeFontNames
	AddIns	Languages
	Application	ListGalleries
	Assistant	MacroContainer
	AutoCaption	NormailTemplate
	AutoCorrect	Options
	CaptionLabels	Parent
	CommandBars	PortraitfontNames
	CustomDictionaries	RecentFiles
	CustomizationContext	Selection
	Dialogs	SynonymInfo
	Documents	System
	FileConverters	Tasks
	FindKey	Templates
	FontNames	VBE
	HangulHanjaDictionaries	Windows
	KeyBindings	WordBasic
PowerPoint 2000	ActivePresentation	CommandBars
	ActiveWindows	Presentations
	AddIns	SlideShowWindows
	Application	Windows
	Assistant	
Access 2000	Application	Forms
	Assistant	Modules
	CodeContextObject	Parent
	CommandBars	References
	DBEngine	Report
	DoCmd	Screen
	FileSearch	
Excel 2000	ActiveCell	MacroSheets
	ActiveChart	Excel14MacroSheets
	ActiveSheet	Names
	ActiveWindow	Parent
	ActiveWorkbook	Range
	AddIns	Rows
	Application	Selection
	Assistant	Sheet
	Cells	ThisWorkbook
	Charts	Windows
	Columns	Workbooks
	CommandBars	WorksheetFunction
	Excel14Intl	Worksheets

全局变量的含义和用途将在以后的内容中介绍，以后可以通过查询帮助文件来具体了解它们的含义和功能。

需要全本请访问 www.ertongbook.com