

宏 ■ 模板 ■ VBA ■ 开发

Excel VBA

入门与实例演练

孙敬杰 / 编著

本书由国内知名 Excel 专家精心编著，强调基础知识与实例演练相结合，既讲解理论知识，又讲解实际应用，帮助读者迅速提高 Excel 使用效率

本书适用于 Excel 2000/2002/2003 多个版本，是 Excel 软件使用者、公司办公人员以及相关专业学生的必备参考书，是读者追求 Excel 更上层楼的绝佳选择

本书特色：

- ◆ 紧密结合日常工作列举 13 大类 76 个 VBA 具体应用实例
- ◆ 侧重介绍 VBA 在样式、图表、报表、分析等方面的应用
- ◆ 集中讲解了大型 Excel VBA 完整实例的制作步骤和方法
- ◆ 兼顾入门级读者和有经验用户对 VBA 学习的各种不同需求
- ◆ 讲练结合的写作模式和课后习题，帮助读者更快速地入门



随书附赠光盘内含各章节所用到的范例数据、过程以及最终完成文件，另外还附赠 Excel 实用模板、升级补丁和相关专题资料，能够帮助广大读者更有效地学习



中国青年出版社

中国青年电子出版社

<http://www.21books.com> <http://www.cgchina.com>

宏 ■ 模板 ■ VBA ■ 开发

Excel VBA

入门与实例演练

孙敬杰 / 编著



中国青年出版社

中国青年电子出版社

<http://www.21books.com> <http://www.cgchina.com>

本书由中国青年出版社独家出版。未经出版者书面许可,任何单位和个人不得以任何形式复制或传播本书的部分或全部内容。

图书在版编目(CIP)数据

Excel VBA 入门与实例演练 / 孙敬杰编著. —北京: 中国青年出版社, 2005

ISBN 7-5006-6157-6

I.E... II.孙... III.电子表格系统, Excel IV. TP391.13

中国版本图书馆 CIP 数据核字 (2005) 第 037788 号

书 名: Excel VBA 入门与实例演练

编 著: 孙敬杰

出版发行: 中国青年出版社

地址: 北京市东四十二条 21 号 邮政编码: 100708

电话: (010) 84015588 传真: (010) 64053266

印 刷: 中国农业出版社印刷厂

开 本: 787 × 1092 1/16 印 张: 24

版 次: 2005 年 6 月北京第 1 版

印 次: 2005 年 6 月第 1 次印刷

书 号: ISBN 7-5006-6157-6/TP · 457

定 价: 39.00 元 (赠 1CD)

前 言

Excel 是 Microsoft Office 家族成员中一个功能强大、技术先进、使用方便的表格式数据综合管理和分析系统。它采用电子表格方式进行数据处理,工作直观方便。Excel 为用户提供了丰富的函数,用户可以随意进行各种数据处理、统计分析和辅助决策,并将其应用于财政、金融、统计、管理等领域。Excel 具有强大的制图功能,可以方便地绘出各种专业图表,实现图、文、表三者的完美结合。最新版本的 Excel 还新增了预防宏病毒的功能、插入其他图片和超级链接的功能(直接链接来自 Internet 或其他网络地址的文档),以及视图管理器、共享工作簿等功能。除上述功能外 Excel 的强大功能还在于它提供的宏语言 VBA。

Visual Basic for Application (简称 VBA) 是新一代标准宏语言,基于 Visual Basic for Windows 发展起来,支持面向对象的程序设计语言。由于它具有直接应用 Office 套装软件的各项强大功能,且其宏记录器可以很容易地将日常工作转换为 VBA,轻松实现办公自动化,因此近年来广受高级办公人员的青睐。但是由于 VBA 控件涉及到程序的编制导致很多办公人员对此望而却步。

作者在翻阅大量同类书籍后发现目前市场上没有一本真正适合办公人员阅读的 VBA 书籍,于是萌发了为最常用 Excel 的办公人员写一本最适合的书籍的念头。作者通过反复的筛选将 Excel VBA 中那些最常用、最基础以及最适用于办公人员的一些功能挑选出来,并通过一个个完整实用的实例将这些 Excel VBA 知识进行了串讲,将那些枯燥无比的编程工作融合到实际的工作中,让广大的办公人员不需要任何的编程基础,通过书中的实例就可以完成 VBA 的学习和应用。如果读者连学习的时间都没有那也没有关系,只要在工作中遇到困难时通过查阅本书,便可立即套用本书中的实例解决问题。

本书的讲解使读者既能够迅速、轻松、高效地完成所面对的各种复杂工作,又不至于陷入学习编程的苦恼中。通过对本书的学习或者查阅,读者可以轻而易举地开发出属于自己的通用程序集或程序块。作者在讲解 Excel VBA 的时候融入了大量的个人开发经验及实例,可帮助读者开发属于自己的高效的通用程序。本书不像一般 Excel 入门书籍那么简单,也不像一般讲解 VBA 书籍那么晦涩,而是将知识点与实际工作相结合,让读者做到即学即用。

作者从普及 Excel 和 VBA 编程知识的角度,介绍 Excel VBA 在办公中的应用,简单讲述了编程技巧,可便于用户提高应用 Excel 的能力,从而达到提高和加深理解宏编程的目的。

全书共 16 章。其中第 1 章和第 2 章讲述 VBA 程序设计语言的基本内容,包括该语言的历史和 VBA 的基本语言要素。第 3~10 章讲述了 VBA 在办公中的基本应用。第 11 章和第 12 章介绍了 VBA 在 Excel 中图表和图像方面的应用。第 13~15 章介绍了 VBA 的高级应用。第 16 章则详细介绍了几个大型的 Excel 系统使用 VBA 设计语言的设计步骤和方法。每章后面都附有小结和习题,供读者加深学习印象。本书随书附赠光盘包含了书中所

讲实例的所有源文件、过程文件及结果文件。

本书主要面对办公室人员、从事办公系统的优化和维护人员以及编程爱好者。对于专业系统开发人员而言，本书也有很高的参考价值。

由于书稿涉及许多专业方面的实例，且基于作者水平和学识所限，加之时间仓促，书中难免存在错误，诚望各位读者不吝赐教斧正，至臻完善。

作 者

2005 年 4 月

目 录

第1章 Excel VBA 简介

1.1 VBA 发展史	1
1.1.1 VBA 历史	1
1.1.2 使用 VBA 的范围	2
1.2 Excel VBA 优势之所在	4
1.2.1 宏	4
1.2.2 基于对象方法	6
1.2.3 VBA 与 VB 的比较	7
1.2.4 VBA 对象的属性及方法	8
1.3 VBA 如何与 Excel 交换信息	9
1.4 本章小结	10
习题	11

第2章 VBA 设计结构

2.1 VBA 开发环境简介	13
2.1.1 VBA 启动方式	13
2.1.2 VBA 操作界面简介	13
2.1.3 VBA 基本概念	15
2.2 VBA 语法基础	16
2.2.1 VBA 字符集和标识字符	16
2.2.2 VBA 常量与变量	17
2.2.3 VBA 数据类型	22
2.3 VBA 运算符	24
2.3.1 算术运算符	24
2.3.2 字符串运算符	26
2.3.3 逻辑运算符	26
2.3.4 关系运算符	30
2.3.5 运算符优先级	31
2.4 VBA 控制语句	31
2.4.1 顺序结构语句	31
2.4.2 循环语句	35
2.4.3 选择语句	37
2.5 VBA 程序结构	41
2.5.1 程序结构说明	41
2.5.2 安全结束宏	44

2.6 本章小结	45
习题	46

第3章 宏与模板

3.1 对基本操作进行宏编辑	47
3.2 对销售情况分别制图的宏	49
3.3 使用 MsgBox 函数查看记录内容	52
3.4 用宏自动计算账簿金额	54
3.5 本章小结	56
习题	58

第4章 构造个性表单

4.1 记录一个自动排序的宏	59
4.2 修改已有的图表宏	61
4.3 制作职工工资卡	64
4.4 优化工资表编制	65
4.5 让表格动起来	68
4.6 使用宏来删除工作表	71
4.7 快速汇总多个工作簿	73
4.8 本章小结	76
习题	79

第5章 单元格样式设计

5.1 修改登记的货物名	81
5.2 将所有货物英文名字全部小写	84
5.3 改变表中出现的所有同名货物	86
5.4 确定金额是否为阿拉伯数字	89
5.5 将货币数字变为中文形式	91
5.6 删除多余空格	95
5.7 插入键盘上没有的特殊字符	98
5.8 真正实现银行利率四舍五入	100
5.9 自定义函数实现四舍六入	103
5.10 本章小结	106
习题	107

第6章 日期和时间

6.1 整理周年庆典日期	109
6.2 计算员工实际使用工作日	111

6.3 给交易加上日期	114	10.2 统计分析数据库	206
6.4 给行程表加上具体时间	116	10.3 项目审计台账管理	209
6.5 确定到下一次会议的时间间隔	118	10.4 通过置信度得到大概到货时间段	215
6.6 确定员工退休年龄方法	121	10.5 根据以前的销售额预测将来 的销售额	217
6.7 自定义会计时间	123	10.6 本章小结	219
6.8 向报表中加入日历	126	习题	221
6.9 本章小结	128	第 11 章 图表的设计及制作	
习题	129	11.1 不同部门年度销售比较	223
第 7 章 查询与统计		11.2 绘制收支平衡图	225
7.1 查找通讯簿中的邮政编码	131	11.3 巧妙构造进度图	229
7.2 获取平均支出或收益	133	11.4 使多个图表同步	233
7.3 搜寻在册员工	135	11.5 使用宏修改图表属性	236
7.4 统计固定资产在当前的折旧值	138	11.6 制作完成比例图的技巧	239
7.5 统计在固定利率下按揭额	140	11.7 本章小结	242
7.6 统计部门中固定年龄段的人数	142	习题	243
7.7 统计投资净现值和收益率	146	第 12 章 图形的动态处理	
7.8 销售结果分析统计	150	12.1 对销售数据制作数据透视表	245
7.9 本章小结	154	12.2 取得公司销售透视表中的数据	248
习题	157	12.3 透视表在预算管理中的应用	253
第 8 章 报表的开发		12.4 对销售数据制作数据透视图	257
8.1 设计只读损益类报表	159	12.5 多重合并计算数据区域	261
8.2 开发实时报表	162	12.6 本章小结	264
8.3 使用窗体录入报表	164	习题	265
8.4 制作明细账簿	168	第 13 章 自定义高级打印设置	
8.5 通过报表模板制作员工卡	171	13.1 打印任意的 Excel 表单	267
8.6 本章小结	175	13.2 打印选取区域	269
习题	176	13.3 打印数据透视表	271
第 9 章 表格修订及权限设置		13.4 设置定时打印	274
9.1 修订表单格式	177	13.5 本章小结	276
9.2 对表单进行快速排序	181	习题	276
9.3 对单元格的格式修订	185	第 14 章 数据的导入及导出	
9.4 为数据加密	189	14.1 从互联网上获取数据库	279
9.5 拒绝错误的贷款支付	193	14.2 将 SQL 数据库转为 Excel 工作表	281
9.6 填充遗漏的单元格	196	14.3 本章小结	283
9.7 本章小结	199	习题	284
习题	201	第 15 章 扩展系统功能	
第 10 章 分析及预测		15.1 宏病毒的原理和防治	285
10.1 构建资产组合投资收益 与风险的优化模型	203	15.2 浮点金额的偏差的解决	290

15.3 工作中也帮你轻松背单词.....	293	16.4 本章小结.....	347
15.4 利用 VBA 创建多级选单.....	296	习题.....	350
15.5 本章小结.....	298	附录 A Excel VBA 的对象	
习题.....	299	附录 B Excel VBA 的事件	
第 16 章 综合实例		附录 C Excel VBA 的属性	
16.1 构建住房公积金管理信息系统.....	301	附录 D Excel VBA 的方法	
16.2 员工工资管理系统.....	316	附录 E 习题答案	
16.3 货物销售管理系统.....	332		

第 1 章 Excel VBA 简介

1.1 VBA 发展史

1.1.1 VBA 历史

VBA 尽管是一个新兴的程序设计语言，但是它的历史几乎与整个计算机工业的发展历史一样长久。VBA 是 Basic 编程语言中较为流行的一种语言（Basic 是 Beginners All purpose Symbolic Instruction Code 的缩写，于 60 年代开始着手开发），它继承了 Basic 的很多特性和优点，更为方便桌面软件设计用户的发展。

近年来，使用 Basic 语言进行程序设计越来越普遍，它所带来的功能也较以前有了很大的提高，并且随着编程语言技术方面的发展和改进，很多软件出版商开始注意到他们的软件应该增强 Basic 方面的功能。所以现在可以从其他许多编程语言（如 Pascal 或 C 等）中发现现代 Basic 语言的许多甚至是全部的特征。所以将 Basic 语言称为高级编程语言的鼻祖一点也不为过。

随着 Basic 语言的发展和提高，应用程序中开始使用宏记录器来记录代码。近年来，宏的应用变得更加复杂以适应用户对宏的要求——功能更灵活，维护更方便。许多宏语言已开始增加那些在复杂的编程语言中才会出现的功能。

为了统一在 Windows 下使用的宏语言，并且可以使用 Windows 的 DDE 和 OLE 方法连接这些应用宏语言，Microsoft 公司创立了一种 Visual Basic 语言的特殊版本，称为 Visual Basic for Application（简称 VBA），Excel 5.0 是第 1 个包含 VBA 的产品。这也是现今所使用的 VBA 的雏形。

要运行一个 VBA 宏程序，必须使用宏保存所在的应用程序。尽管 Excel 可以接收一个从其他的应用程序发来的 DDE 消息并执行一个特定的宏，但是不能通过一个非 Excel 的程序来启动一个 Excel 的 VBA 宏。虽然 VBA 的主要特征在每个应用程序中都大致相同，但是不同的应用程序还是依据不同的需求在 VBA 中增加了特殊的命令和对象。例如，Excel 中的 VBA 有许多命令只与工作表及相关的任务有关，在离开了工作表后，这些宏是无法运行和使用的。

除了增强特定的记录宏以外，用户可以使用 VBA 连接、组排，或控制几个记录宏来完成由几项小任务组成的一项综合性任务。例如，在实际的财会工作中可能经常要从数据库程序中提取数据再输入到一个 Excel 工作表中，将那些要显示的数据格式化，再生成一个图表，并打印出图表和特定格式的报表。为了把这些各自独立的操作通过一个简单的宏组织起来，用户需要分别为每一个操作记录一个宏——一个宏用于输入数据，一个宏用于格式化数据，一个宏用来创建图表，还有一个宏用于打印数据。最后，用 VBA 编写一个简单的宏来组织这些记录宏，并使它们以正确的顺序执行。

在实际应用中也可以使用 VBA 来控制其他应用程序的执行，通过使用 DDE 和 OLE

自动地共享两个或多个应用程序的数据。

VBA 的发展史如图 1-1 所示。

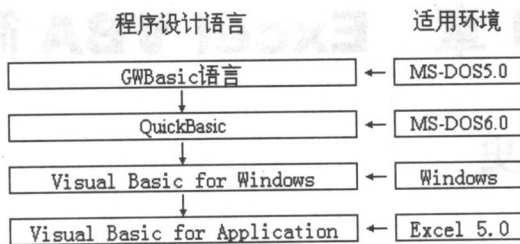


图 1-1 VBA 的发展史

1.1.2 使用 VBA 的范围

虽然 Excel 为用户提供了强大的电子表格功能，但计算机毕竟只是一种帮助人类进行数据处理的机器，因此还有许许多多烦琐的日常工作需要操作者来完成。如工作表数据的输入，工作表格式的设置以及单元格数据计算公式的选择等等。最好的办法就是使用 Excel 中的 VBA 编辑器，因为它不仅功能强大，而且简单易学。简单地讲，在 Excel 中使用 VBA 的益处在于：首先它可以把大量的、重复的操作记录为一个 VBA 程序，用户则只需利用一个简单的菜单命令或工具栏按钮重现程序记录内容，从而实现复杂操作的自动化；其次它可以通过控制 Excel 对象，在一个工作表中灵活地控制其他的工作表、数据表或文档，从而轻松实现不同工作表数据之间的交互；最后利用 VBA 编辑器中提供的丰富的控件设置，用户还可以创建自己的 VBA 应用程序，从而实现一个企业级的信息控制系统。

由于 VBA 与 Excel 结合得非常紧密，因此 Excel 提供的各个功能都可以直接使用 VBA 进行操作。Excel 提供了简单易用的办公软件交互界面，更加容易被广大用户所接受。当需要在 Excel 中实现某些复杂操作或高级功能时，多数用户都会考虑启用 VBA 程序。一般而言，Excel 中 VBA 的适用范围主要包括如下方面。

- 在工作表中完成大量重复操作，例如各种单元格区域的格式化、大量数据的引入等，如图 1-2 所示。

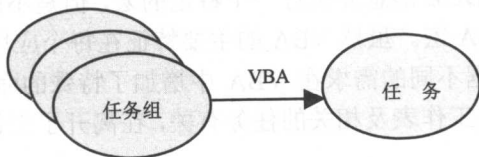


图 1-2 大量重复操作

- 在工作表中完成各种通用操作，例如某种固定类型工作表的创建和删除等，如图 1-3 所示。

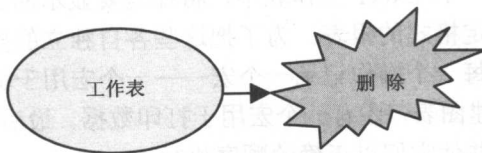


图 1-3 工作表的通用操作

- 在工作表中完成各种交互操作，例如需要用户选择待输入的工作表文件名等，如图 1-4 所示。

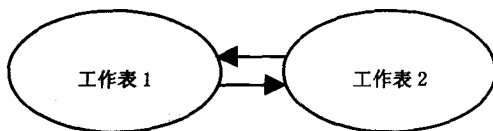


图 1-4 工作表中的交互操作

- 限制工作表中某种数据的范围，例如将某个数据范围限制在数字旋钮中或将某种货物的种类限制在某个列表框所规定的范围内等，如图 1-5 所示。

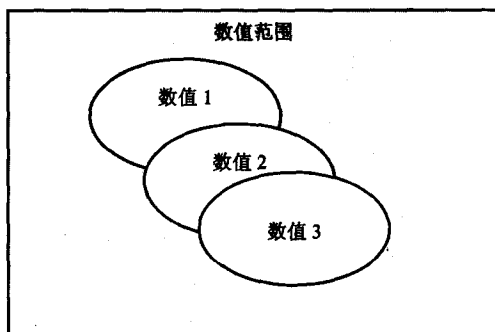


图 1-5 限制数值范围

- 可以利用 Excel 实现一个集成化的信息控制系统，如图 1-6 所示。

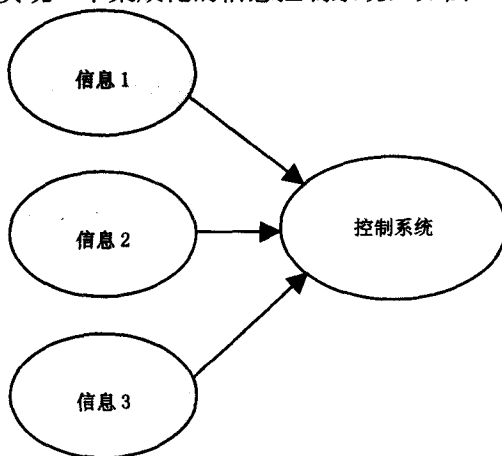


图 1-6 信息控制集成化

除此之外，Excel 中还有很多地方可以使用 VBA 程序。但是，在工作簿中漫无目的地创建大量 VBA 程序并不是一种良好的习惯，这样可能会导致一般用户难于理解指定 VBA 程序模块的功能，而不能读懂程序，从而降低整个工作簿文件的实用性。而且，VBA 的功能虽然强大，但也不是万能的，只有在恰当的地方使用 VBA 程序，才能真正起到事半功倍的效果。

1.2 Excel VBA 优势之所在

1.2.1 宏

在开始学习编写宏之前，用户需要很好地理解什么是宏，以及用 VBA 编程的宏与一个记录的宏有何不同。

宏是一个指令集合，它可以使 Excel 自动地完成用户指定的任何动作组合，而且宏的录制和使用方法相对而言也比较简单，它并不要求用户有任何编程经验或 Visual Basic 编辑器使用经验。因此，用户普遍都比较喜欢在 Excel 中录制并使用宏命令。录制宏命令时，Excel 将自动记录并存储用户所执行的一系列菜单命令的信息。运行宏命令时，Excel 则自动将已经录制的命令组合重复执行或“回放”，从而实现重复操作的自动化。当然，宏命令本身实际上就是一种 VBA 程序，它是存储在 VBA 模块中的一系列命令和函数。当执行宏命令所对应的任务组合时，Excel 将自动启动该 VBA 程序模块的运行程序，VBA 程序的强大功能由此可见一斑。

当一个宏记录了用户想要进行的全部操作——包括错误和错误点，而当一个应用程序再次执行一个宏时，它以准确的顺序执行用户上次执行的命令。如果宏记录中有一个严重的错误，例如包含一个引用错误，那么改正这个错误的惟一办法就是重新记录这个宏。同样，如果要在一个很长的宏中修改一个引用，用户也不得不重新记录整个宏，可是在重新记录新宏的过程中又有可能引发新的错误。正因为这个原因，软件开发者在宏记录器中增加了编辑宏的功能，从而使程序员可以很方便地改正错误或进行其他变动而无需重新记录宏。图 1-7 所示显示了当前工作表中的宏，用户可以单击“编辑”按钮来修正宏。

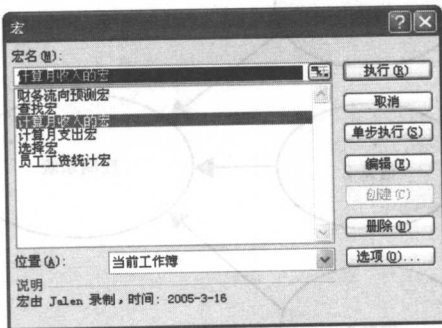


图 1-7 当前工作表包含的宏

“录制新宏”对话框的组件说明如下所示。

- 宏名：在此指定宏的名字。宏名必须以英文字母开头，可以包含字符和数字，但不能包含空格和符号，最大长度为 80。
- 将宏指定到：在此指定运行宏的位置。可以从工具栏上的图表运行宏，还可以采用快捷键运行宏。
- 将宏保存到：可以把要录制的宏存储在宿主应用程序的一个实例中。
- 说明：在此输入与宏相关的详细信息。可以包括一些附加信息，用来帮助其他用户了解如何使用宏。

录制宏的步骤如下所示。

步骤 1 单击 Excel 主菜单上的“工具>宏>录制新宏”选项，如图 1-8 所示。

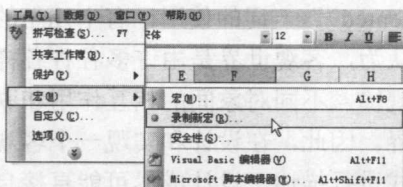


图 1-8 单击“录制新宏”选项

步骤 2 输入为宏指定的名称，如图 1-9 所示。

步骤 3 单击“确定”按钮开始录制。

现在就可以手动设置格式或进行操作了。当开始录制宏时，录制工具栏将只包含两个按钮，即“停止录制”按钮和“相对引用”按钮，右击“相对引用”按钮，在弹出的菜单中包括“暂停录制”选项，如图 1-10 所示。

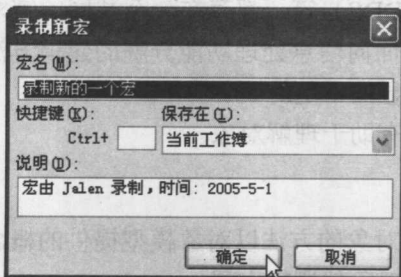


图 1-9 为宏输入指定的名字



图 1-10 “停止录制”按钮

在完成了宏的录制之后，单击“停止录制”按钮结束录制。当录制了一个宏时，宿主应用程序就会为这个宏生成相应的 VBA 代码。可以使用“宏”对话框查看该宏的 VBA 代码。要查看宏的 VBA 代码，可以按如下操作步骤进行操作。

步骤 1 单击 Excel 主菜单中的“工具>宏>宏”选项。

步骤 2 在列表框中选择一个宏，如图 1-11 所示。

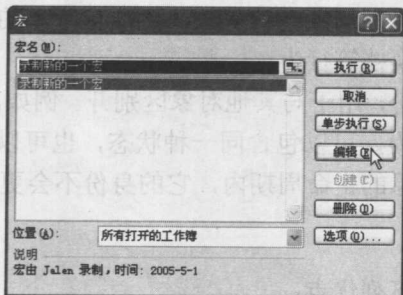


图 1-11 选择宏可以查看其代码

步骤 3 单击“编辑”按钮，就可以查看该宏的 VBA 代码。

总的来说，宏是能够自动完成某个任务的一组指令，这些指令放在一起就如同一个命令一样。宏与 DOS 中的批处理文件类似。批处理文件是用于自动化用户完成的一组任务。

1.2.2 基于对象方法

“面向对象 (Object Oriented)”一词的使用已有十几年的历史,和人们认识世界的规律一样,面向对象的方法学认为,客观世界是由许多各种各样的对象所组成的,每种对象都有各自的内部状况和运动规律,不同对象间的相互作用和联系构成了各种不同的系统,构成了现在所面对的客观世界。因此,在设计和实现一个客观系统时,应按照人们在现实世界中对客观事物的思维方式进行描述,设计成尽可能直接自然表现问题的求解,即寻求一种回归自然的设计思想和方法。面向对象方法所追求的是现实问题空间与软件系统解空间的近似和直接模拟,它希望人们最省力并且最大限度地利用软件系统来求解问题,将客观世界的对象映射到面向对象系统的解空间是面向对象方法的核心内容。

现在,面向对象的研究遍及计算机系统的各个领域,如面向对象的程序设计语言、面向对象的程序设计方法、面向对象的操作系统、面向对象的数据库管理系统以及面向对象的分析方法等。在开发任何一个系统的构成中,无论是会计电子数据处理系统(EDP)、财务管理信息系统(FMIS)和会计决策支持系统(FDDS)等,都存在一个共同的需求,这就是在系统分析阶段建立系统的信息模型。而传统面向信息处理功能分解的建模方法,特别是在建立较大规模财务系统的模型时,存在着明显的不足。

下面将讨论基于对象方法的几个要点,它们将有助于理解对象。

1. 基于对象的方法

对象模型形成了基于对象的方法的基础。基于对象的方法以对象模型提供的概念框架为基础。在基于对象的应用程序中,对象模型是指对象的逻辑布局。

对象是有形的实体,它可以展示一些定义明确的行为。例如,足球是一个对象,它是有形的、可见的实体,并带有可见的边界。然而,并不要求所有对象都包含物理边界。一个组织没有物理边界,而只包含概念上的边界,那么该组织也是对象。

对象具有状态,它可以展示一种行为,而且还具有惟一的身份。

对象的状态由一组属性及其值指定。例如,自行车可能包含很多状态,如两个车轮、多个齿轮和颜色等。

行为是这些属性在某个时间周期内的更改。比如自行车的示例,操作者可以利用自行车来展示诸如刹车、加速和换档等行为。

每个对象都有惟一的身份,用于与其他对象区别开。例如,动物可以用它的种类标识。两个对象可能包含同一种行为,可以包含同一种状态,也可以不包含同一种状态,但从来不会包含同一种身份。在对象的生命周期内,它的身份不会更改。

2. 基于对象方法的优点

基于对象的方法提供了下列优点:

实际的模拟。这是指模拟周围现实世界的对象,换句话说,人们生活在对象的世界里,基于对象的方法模拟了现实世界。

可重用性。这是指把过程保存为外部文件,以便其他应用程序使用。可重用性的优点是能够节省时间和精力,这样会降低费用。

更改的弹性限度。这是指软件在一段时间周期内能够容纳更多对象的灵活性。

如图 1-12 所示，可以看到模拟对象都是实际对象的映射。

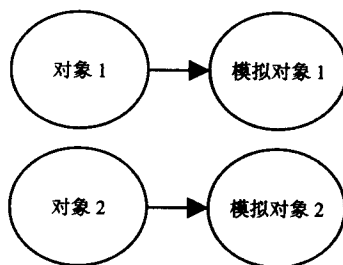


图 1-12 模拟对象都是通过实际对象继承得到

1.2.3 VBA 与 VB 的比较

虽然 VBA 起源于 VB，但它成为一个对应用程序几乎没有束缚的独立工具。VBA 不仅作为一个宏语言挤进每个应用程序，它还保留了一个可以在应用程序之间操作的开放编程工具用来作为一个结合层。

用 Visual Basic for Application 编写的应用程序可以复制到 Visual Basic 中，只需稍加修改，就可以在 VB 中编译，作为 VB 程序运行，用来控制其他应用程序。

由于 VBA 是新一代 Microsoft Windows 应用软件通用的宏控制语言，不但语法简单而且又能直接运用相关软件（如 Excel, Access, PowerPoint 等）的各项强大功能。一旦掌握 VBA，使用者就可以在为最终用户定制的应用软件中直接运用上述各项软件的功能，而不再费时去自行开发。

VBA 除了实用性特别高之外，还能以记录的方式直接将使用者想要的操作记录成程序代码。所以 VBA 实际上也具备了应用程序生成器的特点和功能。

除了针对不同软件功能的综合应用提供更具体的可行方案，VBA 实际上还附带着一些高级应用。VBA 承担了扩展 OLE 的任务，能像交换数据那样交换命令，这些扩展被称为 OLE Automation，并在 OLE-2.0 中发布。Excel 是第一个完全采用 OLE Automation 的应用程序。有了 OLE Automation 的帮助，VBA 对于所有已经会用 Office 套件的用户来说，更是不可多得的好帮手。因为他们马上就能使用 VBA 来应付实践中遇到的特殊需求，利用记录 VBA 程序代码的方式，任何人都可以完成一套应用系统。能直接运用 Excel 来生成数据分析图表，配合 Word 来编辑业务报告，再通过 PowerPoint 来设计出高品质的动态简报等。这些都不再是梦想，掌握了 VBA，想成为 Windows 上 Microsoft Office 的应用大师是轻而易举的。

VBA 与 VB 的开发环境相似。它们来源于同一种编程语言，即 Basic。两种语言包含的对象和方法大致相同。换句话说，对于 VB 所支持对象的多数属性和方法，VBA 也支持。然而，在事件或属性的特定名字方面，会有些许的差别。虽然 VB 及 VBA 都有一个 Basic，但是千万可别只把它当成 Basic 语言的 Visual 版本来看。至于 VBA 与 VB 到底应当如何区分呢？VB 是一种程序语言，而 VBA 则不能被归类于“程序语言”。虽然它具有十分完整的程序语言基本结构（与 VB 十分类似），但它提供的不仅仅是程序代码，而是对 Microsoft Office 各种软件功能的一种综合控制。因此，VBA 和 VB 之间的不同之处是不能在一个环

境中独立运行 VBA，也不能使用它创建独立的应用程序。然而，可以使用 VB 创建工具，用它们创建独立组件，例如可执行的程序和 COM 组件。当从头开始建立应用程序时，VB 尤其有用。

VBA 是完整的编程语言，它可以与那些与之集成的应用程序一起使用。换句话说，VBA 需要宿主应用程序支持它的功能特性。宿主应用程序是诸如 Word、Excel 或 Access 这样的应用程序，它们能够为 VBA 编程提供集成开发环境。宿主应用程序包含很多对象，VBA 可以使用这些对象定制它们。例如，Word 包含自己的一组 VBA 对象，Excel 也包含自己的一组 VBA 对象。只有打开宿主应用程序才能运行 VBA。因此，使用 VBA 不能建立独立的应用程序，但是，通过隐藏宿主应用程序，可以显示 Office 应用程序中创建的用户窗体。这就产生了独立应用程序的假象。因此，VBA 最适用于定制已有的应用程序。

实际上，VBA 的强大开发功能与易用性相结合，实现了从定型软件应用程序到“软件构造模块”的转变。开发人员可以用它们快速生成大量的用户解决方案，将桌面系统与商业和工业领域的特定应用程序、数据和解决方案集成在一起。开发人员可以定制并扩展这些应用程序的功能，以便符合特定业务需求，而不需要从最低层开始开发应用程序，使得 Office 不仅仅是一个桌面办公系统，更确切地讲是一个易学易用的开发平台。

1.2.4 VBA 对象的属性及方法

为了更加方便地管理数据和代码，VBA 全面采用了面向对象的编程技术，对象是封装有数据和代码的客体。在 Office 中有许多对象，例如菜单栏、工具栏、Excel 工作簿、Excel 工作表、窗体、图形及图表等，这些对象相互关联，存在包含与被包含的关系。对象有自己的属性和方法，属性是封装在对象内部的数据，它反映了对象的特征。例如，Excel 的工作表对象有 Caption 和 Cells 等属性。VBA 中每个对象都拥有许多属性，几乎所有对象的属性都是可读的，但不是所有属性都是可写的。对象的方法是封装在对象中用来操作对象属性的代码段。这些代码段对于用户不可读，也不可改写。用户在使用 VBA 对象的方法时，只需关心方法的名称和参数，无需关心方法的具体实现过程。如 Excel 中 Worksheets 对象的 Change 和 Init 方法。

可以利用对象编程来创建解决方案，利用 Office 中提供的对象，无需编写很多代码就可以创建自定义解决方案。例如，可以利用在 Excel 中用 Chart 对象创建一个解决方案，把数据显示给用户。甚至不必从头编写代码来建立图表，就可以直接访问 Chart 对象中已经内建的功能。也可以在很多开发工具中编写代码来控制 Office 的对象，即 OLE-VBA 技术。例如，可以在 Excel 中编写一个过程来改写一个指定 Word 文档中某一个表格的内容，或者在 VBA 中编写应用程序改写或填充 Excel 中某一单元格的内容或格式。

在 VBA 中，要访问一个对象，必须清楚该对象在对象体系中处于何种位置，然后通过对象访问运算符“.”，从包含该对象的最外层开始，由外及里逐次取其子对象，一直到达要访问的对象为止。例如在 Excel 中，要访问当前工作表第 1 列第 2 行的单元格的存储内容，应按下列格式书写：

```
Application.ActiveSheets.Range("A2").Value
```

熟悉 VBA 开发环境是利用 VBA 编程的首要条件，只有掌握了 VBA 开发环境和编程

的一些基本概念,才能顺利的编写 VBA 程序。Microsoft Office 的所有应用程序都拥有共同的开发环境,即 Visual Basic 编辑器,该编辑器与 VB 编辑器基本相同,无论在 Word 还是在 Excel, PowerPoint 等应用软件中,都拥有自己的“Visual Basic 编辑器”。

Visual Basic 编辑器提供了与其他 Microsoft 可视化编程工具相似的开发环境,用户可以快速在整个工程中定位、编辑和运行代码。窗体设计器使每个用户能够建立包含 ActiveX 控件的自定义对话框。它还包含完整的调试工具,用户可以选择单步执行代码、设置断点和监视点等。

Visual Basic 编辑器中包含一个“工程资源管理器”,它显示了当前工程中包含的各种元素,其中包含应用程序的对象、窗体、模块、类和引用的其他文档、模板和外接程序等,通过树型结构显示,大家可以直观地看到 VBA 工程元素的层次关系。

在 Office 中,VBA 的对象很多,对象的方法和属性更为繁杂。如果用户不熟悉 VBA 的对象以及对象的属性和方法,几乎不可能写出有效的程序代码。为了使用户快速学习和掌握 VBA,特别是帮助用户快速构造 VBA 使用程序,首先应考虑使用宏记录器记录 VBA 代码的主要功能和执行步骤,然后根据实际需要进行编辑改造,这不仅对于使用 VBA 开发应用系统的程序设计人员来说是首选途径,也是 VBA 编程一个很好的方法。

1.3 VBA 如何与 Excel 交换信息

Excel 的众多对象之间相互关联并构成了一个有机的整体。其中 Application 对象位于最上层,代表了整个 Excel 应用程序。包括应用程序的各种选项以及当前活动的各种对象。其下面主要有 Workbooks (工作簿集合)对象、Window (窗口)对象、Dialog (对话框)对象等。而 Workbooks 对象下面又有 Worksheets (工作表)对象、Chart (图表)对象、Window (窗口)对象、Name (名称)对象等。Worksheets 对象下面还有 Range (区域)对象、Name (名称)对象等。其中,Application 对象下面的 Window 对象与 Workbook 对象下面的 Window 对象是不同的。前者包含了 Excel 的所有窗口,而后者只包含指定工作簿中的窗口。

不同的对象通常具有不同的方法、属性和事件。例如 Workbooks 对象有 Add, Close, Open 等方法,Application, Count, Item, Parent 等属性。集合型对象都有 Count, Item 属性,可得到相应集合中的对象个数或指定的对象。而除了 Application 对象以外的所有的对象都有 Parent 属性,可得到相应对象的父对象。不同对象包含的事件也是不同的。例如 Worksheet 对象就有 Activate, Deactivate, Change 等 7 个事件,而 Workbook 对象则包含了多达 19 个事件。

VBA 主要是根据用户对不同对象的操作以及触发的相应的事件去激活相应的程序,通过对相关的对象执行不同的方法,或是修改相关对象的属性来完成特定的工作。举例说明,首先看下面的一段 VBA 程序。

```
Sub Example()  
    Range("A1:F1").Select  
    Selection.NumberFormat = "abcdefg"  
    ActiveWindow.DisplayGridlines = False  
End Sub
```