

Excel 2013 VBA

入门与应用

复杂数据简单化 · 分析处理自动化 · 日常办公高效化



分析 可视
发现 共享



覆盖Excel VBA所有基础知识点和对象编程技巧

郭刚 编著

循序渐进，由浅入深，实例丰富，应用为先

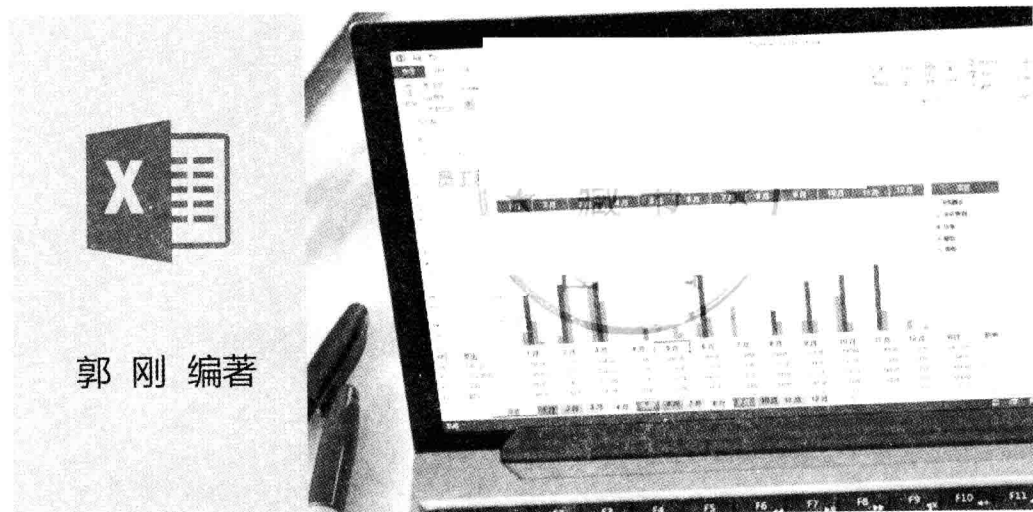


本书示例文件、PPT教学课件与教学视频

清华大学出版社

Excel 2013 VBA

入门与应用



清华大学出版社

内 容 简 介

本书向读者介绍 Excel VBA 程序设计的基础知识,展示 VBA 编程的实战技巧,帮助读者快速经历 Excel VBA 从入门到精通的成长过程。

全书共 15 章,内容涵盖了 Excel 的宏、Excel VBA 的开发环境、VBA 语言基础、VBA 的基本语句、VBA 的语句结构、过程和函数的概念、对象的知识、常用对象的使用(包括 Application 对象、WorkBook 对象、Range 对象、WorkSheets 对象、Chart 对象和 Shape 对象)、数据操作的技巧。本书最后提供了一个员工信息管理系统的综合案例,可让读者综合所学知识进行应用开发。随书配送的多媒体教学光盘,包括书中全部实例的操作视频、源文件和 PPT 文件,可随时进行调用和播放学习。

本书适合需要学习 Excel VBA 程序设计的初级用户,以及希望提高 Excel VBA 编程能力的中级用户阅读,同时也适合于相关专业的高等院校师生学习阅读。

本书封面贴有清华大学出版社防伪标签,无标签者不得销售。

版权所有,侵权必究。侵权举报电话:010-62782989 13701121933

图书在版编目(CIP)数据

Excel 2013 VBA 入门与应用 / 郭刚编著. - 北京:清华大学出版社, 2014

ISBN 978-7-302-35340-9

I. ①E… II. ①郭… III. ①表处理软件 IV.①TP391.13

中国版本图书馆 CIP 数据核字(2014)第 021096 号

责任编辑:夏非彼

封面设计:王 翔

责任校对:闫秀华

责任印制:李红英

出版发行:清华大学出版社

网 址: <http://www.tup.com.cn>, <http://www.wqbook.com>

地 址:北京清华大学学研大厦 A 座

邮 编:100084

社 总 机:010-62770175

邮 购:010-62786544

投稿与读者服务:010-62776969, c-service@tup.tsinghua.edu.cn

质 量 反 馈:010-62772015, zhiliang@tup.tsinghua.edu.cn

印 刷 者:清华大学印刷厂

装 订 者:三河市漂源装订厂

经 销:全国新华书店

开 本:190mm×260mm

印 张:28

字 数:717 千字

附光盘 1 张

版 次:2014 年 5 月第 1 版

印 次:2014 年 5 月第 1 次印刷

印 数:1~3000

定 价:65.00 元

产品编号:055068-01



前言

Excel 是一款功能强大、技术先进且使用方便的数据分析和管理软件。Excel 的优秀表现，除了其强大的制表功能之外，还在于其具有二次开发的能力。Excel 允许用户以其作为平台开发新的工具，以完成 Excel 本身所不具有的功能。实现 Excel 二次开发的工具就是内置于 Excel 中的 VBA 语言，通过 VBA 来进行 Excel 应用程序的开发，能够增强 Excel 的自动化水平，提高 Excel 完成日常工作的效率，实现对复杂数据处理的简化。

对于非计算机专业的读者来说，学习一门计算机编程语言并非一件容易的事情，笔者也曾经历过 VBA 初学时的挣扎和入门后的迷茫，能深深体会到一本兼顾知识性和实用性的参考书对于学习 VBA 的意义。为了帮助广大读者快速掌握 VBA 程序设计的特点，轻松提高程序设计的能力，笔者根据自己多年学习和应用 Excel VBA 所获得的经验和体会编写了本书。本书从 Excel VBA 程序设计的基础知识开始，引领读者一步步深入了解 VBA 程序设计的应用。全书围绕在 VBA 应用中遇到的各种问题来进行讲解，可帮助读者快速掌握 Excel VBA 程序设计的相关知识并获得技能。

本书特点

1. 内容充实，知识全面

本书从 Excel VBA 的基本语法知识出发，介绍了 Excel VBA 中对象编程技巧、Excel 应用程序界面的制作、Excel VBA 与数据库的应用以及 Excel 与外部文件的交互等内容，涵盖了 Excel VBA 应用程序开发的方方面面，内容全面而丰富。

2. 循序渐进，由浅入深

本书面向 Excel VBA 各个层面的用户，以帮助读者快速掌握 Excel VBA 程序设计为目标。全书在结构上采用由浅入深的方式，从读者学习的角度出发，以解决读者在学习过程中遇到的问题和掌握的使用技能为己任。在内容安排上，层层推进，步步深入，让读者实现“由入门到精通，由知之到用之”的平滑过渡。

3. 实例丰富，实用为先

理解概念，掌握技巧，离不开编程实例。本书提供了大量的实例，实例选择力求具有针对性，与知识点紧密结合并突出应用技巧。实例在设计上不追求高精尖，突出实用性，以利于读者理解和实际操作。

4. 优化代码，深入剖析

本书实例代码短小精悍，使用的算法不求高深，易于运行。本书没有复杂的理论讲解，通过代码来体现知识的应用技巧，力求以最简单的语句来解决最实际的问题。



5. 类比讲解，描述直观

本书在对 VBA 对象方法和属性进行介绍时，以 Excel 操作来进行类比，帮助读者快速理解它们的作用。同时，全书图文丰富，以直观的描述方式来将知识要点和程序运行特征呈现到读者面前。

6. 适用性强，便于速查

本书介绍的所有知识、编程方法和技巧同样适用于 Excel 的早期版本，如 Excel 2003 和 Excel 2010。书中涉及到的大多数源代码均可以在实际应用中直接使用。同时，本书采用应用驱动模式，用户可以通过目录快速查找需要的操作任务实例，方便学习。

本书结构

本书共 15 章，章节内容如下：

第 1 章介绍 Excel 宏的有关知识，主要包括录制宏的方法、设置宏的启动方式、了解加载宏和 Excel 中的宏安全设置等知识。

第 2 章介绍 VBA 的开发环境，包括认识 VBA 编辑器、了解 VBA 编辑器中常见窗口和在 VBA 编辑器中输入代码的技巧。

第 3~8 章介绍 VBA 程序设计的基础知识，包括 VBA 的数据类型和运算符、VBA 的常用语句、VBA 程序的流程控制、VBA 过程和函数的概念以及 VBA 中对象的知识。

第 9~14 章介绍 Excel VBA 中常见对象的使用，包括 Application 对象、WorkBook 对象、Worksheets 对象、Range 对象、Chart 对象和 Shape 对象。

第 15 章是一个综合案例，介绍了使用 Excel VBA 制作用于对企业员工信息进行管理的实用系统的过程。

本书读者

- 需要获得 Excel VBA 程序设计入门知识的零基础读者
- 初步掌握 Excel VBA 知识，需要进一步提高的初学者
- 需要通过 VBA 提高 Excel 数据处理效率的各类从业人员
- 高等院校学生和相关授课教师
- 企业或相关单位培训班学员

编 者

2014 年 2 月

目 录

第 1 章 迈向 VBA 的第一步从宏开始	1
1.1 用宏记录操作	1
1.1.1 录制宏	1
1.1.2 宏与 VBA	3
1.2 宏的运行方式	4
1.2.1 使用“宏”对话框运行宏	4
1.2.2 使用快捷键运行宏	5
1.2.3 使用对象运行宏	6
1.2.4 从快速访问工具栏运行宏	6
1.2.5 从选项卡运行宏	8
1.3 加载宏的应用	10
1.3.1 使用 Excel 的加载宏	10
1.3.2 录制加载宏	12
1.3.3 卸载加载宏	14
1.4 宏为什么运行不了	15
1.4.1 为宏的使用上一把锁	15
1.4.2 对文件进行限制	17
第 2 章 了解 VBA 开发环境	19
2.1 初识 VBE	19
2.1.1 如何进入 VBE 环境	19
2.1.2 Visual Basic 编辑器的构成	21
2.2 Visual Basic 编辑器中的窗口	22
2.2.1 工程资源管理器	22
2.2.2 “属性”窗口	23
2.2.3 “代码”窗口	25
2.2.4 立即窗口	27
2.3 VBA 代码输入很简单	27
2.3.1 代码输入很简单	27
2.3.2 如何查询对象的属性和方法	29
2.3.3 获得帮助很容易	30



第 3 章 从零开始学习 VBA	32
3.1 了解 VBA 的数据类型	32
3.1.1 认识 VBA 基本数据类型	32
3.1.2 特殊的枚举类型	34
3.1.3 数据类型是可以自定义的	35
3.2 存储数据的容器：变量	36
3.2.1 使用变量的第一步：声明变量	36
3.2.2 先声明，再使用——强制声明变量	37
3.2.3 变量的作用域	38
3.2.4 变量的生存周期	39
3.3 必不可少的常量	40
3.3.1 直接常量	41
3.3.2 符号常量	41
3.3.3 系统常量	42
3.4 告诉程序怎么算	43
3.4.1 进行计算的算术运算符	43
3.4.2 比较大小的比较运算符	44
3.4.3 进行逻辑运算的逻辑运算符	44
3.4.4 合并字符的连接运算符	45
3.4.5 应该先算什么	46
3.5 大量数据的操作从数组开始	47
3.5.1 初识数组	47
3.5.2 声明数组	48
3.5.3 随心所欲的动态数组	49
第 4 章 编写程序，从语句开始	51
4.1 编写程序代码的规则	51
4.1.1 编写代码必须遵循的原则	51
4.1.2 语句很长和很短怎么办	52
4.2 最基本的语句——赋值和注释语句	53
4.2.1 赋值语句	53
4.2.2 注释语句	54
4.3 输入输出何须控件	55
4.3.1 使用输入对话框输入数据	55
4.3.2 使用 MsgBox 函数输出数据	57
4.3.3 简单实用的 Print 方法	60

4.4 暂停和退出程序	62
4.4.1 让程序暂时停止一下——使用 Stop 语句	62
4.4.2 停止程序的运行——End 语句	64
第 5 章 控制程序的流程——VBA 的基本语句结构	66
5.1 何必一条道跑到黑——VBA 的分支结构	66
5.1.1 实现单一条件选择	66
5.1.2 实现双重条件选择	69
5.1.3 实现多重选择	71
5.1.4 特殊的多分支语句	73
5.2 重复, 很适合程序的方式——使用循环结构	76
5.2.1 指定循环次数的循环	76
5.2.2 针对数组和对象集合的循环	78
5.2.3 先条件后循环	80
5.2.4 先循环后条件	82
5.3 层层嵌套功能强	84
5.3.1 分支结构的嵌套	84
5.3.2 循环结构的嵌套	86
5.4 找出程序中的错误	88
5.4.1 让程序跳转到指定位置	89
5.4.2 抓住程序中的错误	90
5.4.3 错误处理完了该怎么办	92
第 6 章 过程, VBA 的基本程序单元	95
6.1 初识过程	95
6.1.1 过程放在哪里	95
6.1.2 VBA 包含哪些过程	96
6.2 使用 Sub 过程二三事	97
6.2.1 如何创建 Sub 过程	98
6.2.2 过程间的互相调用	101
6.2.3 过程的作用域	103
6.3 你有我有全都有——传递参数	104
6.3.1 传递参数的两种方式	104
6.3.2 数组也可以作为参数	107
6.3.3 参数没有传递该怎么办	108
6.3.4 参数的数量无法确定该怎么办	110



第 7 章 过程的孪生兄弟——函数	112
7.1 有了过程为什么还要函数	112
7.2 使用函数很简单	113
7.2.1 创建 Function 函数	113
7.2.2 在工作表中调用 Function 函数	114
7.2.3 在 VBA 程序中调用函数	117
7.2.4 在程序中使用工作表函数	119
7.3 内置函数很方便	121
7.3.1 使用判断函数	121
7.3.2 使用日期/时间函数	124
7.3.3 使用字符串函数	125
7.3.4 使用转换函数	127
7.3.5 使用算术函数	128
第 8 章 对象，控制 Excel 的基础	131
8.1 对象三要素	131
8.1.1 对象的属性	131
8.1.2 对象的方法	133
8.1.3 对象事件	135
8.2 拿什么来装载你，我的对象	137
8.2.1 对象变量	137
8.2.2 对象数组	139
8.3 Excel 的对象模型	141
8.3.1 Excel 的对象模型	141
8.3.2 如何引用对象成员	142
8.3.3 什么是对象集合	144
第 9 章 位于顶层的 Application 对象	147
9.1 对 Excel 进行梳妆打扮	147
9.1.1 改变 Excel 窗口的位置	147
9.1.2 设置 Excel 窗口标题文字	148
9.1.3 设置状态栏和编辑栏	150
9.1.4 设置鼠标指针形状	151
9.2 对文件进行操作	153
9.2.1 获取文件名	153
9.2.2 获取文件的保存位置	155

9.2.3	打开文件	157
9.2.4	打开最近使用的文档	158
9.3	操作 Excel	159
9.3.1	了解 Excel	159
9.3.2	对“最近使用的工作簿”列表进行操作	161
9.3.3	设置保存自动恢复文件的时间间隔和保存位置	162
9.3.4	使 Excel 不显示警告信息对话框	163
9.3.5	设置新工作簿中工作表的个数	164
9.3.6	为过程的启动指定快捷键	166
9.3.7	实现定时操作	167
9.3.8	退出 Excel 应用程序	168
9.4	与单元格有关的操作	169
9.4.1	取消对单元格的复制或剪切操作	170
9.4.2	设置单元格的选择方向	171
9.4.3	控制函数名称列表的显示	172
9.4.4	设置编辑栏的高度	173
9.4.5	控制浮动工具栏的显示	175
9.4.6	设置多线程计算	176
9.4.7	在工作表中快速选择单元格	178
9.4.8	同时选择多个单元格区域	179
9.5	使用对话框	180
9.5.1	使用内置对话框	180
9.5.2	使用输入对话框	182
9.6	应用程序级的事件——Application 事件	184
9.6.1	如何使用 Application 事件	184
9.6.2	激活工作表时触发的事件	186
9.6.3	激活工作簿时触发的事件	187
9.6.4	关闭工作簿时触发的事件	188
9.6.5	在更改窗口大小时触发的事件	189
第 10 章	管理工作簿，从 Workbook 对象开始	191
10.1	使用它，先了解它	191
10.1.1	引用工作簿	191
10.1.2	判断工作簿是否打开	193
10.1.3	判断工作簿是否已经保存	194
10.1.4	获取工作簿的属性信息	195



10.1.5	获取工作簿的名称和完整路径	197
10.2	操作工作簿	198
10.2.1	打开工作簿	198
10.2.2	保存工作簿	200
10.2.3	创建新的工作簿	203
10.2.4	关闭工作簿	204
10.2.5	为工作簿添加打开密码	205
10.2.6	保护工作簿	206
10.2.7	将工作簿发布为 PDF 文件	208
10.3	操作工作簿窗口	211
10.3.1	引用工作簿窗口	211
10.3.2	设置工作簿窗口的显示状态	212
10.3.3	设置工作簿窗口的显示比例	214
10.3.4	让单元格在工作簿窗口左上角显示	215
10.3.5	设置网格线的颜色	216
10.3.6	拆分窗口并冻结窗格	218
10.4	使用工作簿事件	219
10.4.1	在打开工作簿时触发的事件	219
10.4.2	在激活工作簿时执行程序	221
10.4.3	保存工作簿之前触发的事件	222
10.4.4	关闭工作簿之前触发的事件	223
10.4.5	工作簿处于非活动状态时触发的事件	224
10.4.6	新建工作表时触发的事件	225
10.4.7	工作表被激活时触发的事件	227
10.4.8	当工作表中单元格数据发生改变时触发的事件	228
10.4.9	双击工作表时触发的事件	228
10.4.10	用右键单击工作表时触发的事件	230

第 11 章 操作工作表，使用 Worksheets 对象 231

11.1	名称和索引号，引用的关键	231
11.1.1	使用索引号引用工作表	231
11.1.2	使用名称引用工作簿	233
11.1.3	Sheets 对象	234
11.2	操作工作表	236
11.2.1	新建工作表	236
11.2.2	删除工作表	238

11.2.3	选择工作表	240
11.2.4	复制工作表	241
11.2.5	移动工作表	243
11.2.6	保护工作表	245
11.2.7	打印工作表	247
11.2.8	隐藏工作表	248
11.3	事件, 可以方便实现与工作表的交互	251
11.3.1	激活工作表时触发的事件	251
11.3.2	单元格数据发生变化时触发的事件	252
11.3.3	选择区域发生变化时触发的事件	254
11.3.4	重新计算时触发的事件	255
11.3.5	双击工作表时触发的事件	256
第 12 章	数据操作的核心, 使用 Range 对象	258
12.1	对号入座, 获取单元格对象	258
12.1.1	使用 Range 属性实现引用	258
12.1.2	使用 Cells 属性	261
12.1.3	引用行列	264
12.1.4	用偏移量来实现引用	265
12.1.5	缩放单元格区域	267
12.2	合纵连横, 引用单元格区域	268
12.2.1	引用工作表的内容区域	268
12.2.2	获取内容区域的开头和结尾	271
12.2.3	引用多个非连续单元格区域	272
12.2.4	引用单元格区域的交叉区域	273
12.2.5	引用单元格区域中的不连续区域	275
12.3	设置单元格外观	276
12.3.1	设置单元格边框	276
12.3.2	对单元格进行填充	281
12.3.3	设置单元格中文字格式	282
12.3.4	设置单元格的大小	284
12.3.5	使用自动套用格式	285
12.3.6	设置单元格数据格式	287
12.4	操作单元格	289
12.4.1	删除单元格	289
12.4.2	插入单元格	291



12.4.3	复制和粘贴操作	292
12.4.4	合并和拆分单元格	295
12.4.5	保护单元格	298
12.4.6	清除单元格内容	300
第 13 章 操作数据得心应手		302
13.1	找到需要的数据	302
13.1.1	查找特殊单元格	302
13.1.2	查找单个符合条件的数据	305
13.1.3	查找多个符合条件的数据	308
13.1.4	替换数据	310
13.1.5	查找具有特定格式的单元格	312
13.1.6	实现模糊查询	314
13.1.7	使用内置函数	316
13.2	排排坐，吃果果——数据的排序	318
13.2.1	对数据进行排序	319
13.2.2	按照自定义序列排序	322
13.2.3	按照颜色排序	325
13.2.4	按照单元格中字符的长度排序	327
13.2.5	对同时包含数字和字母的单元格进行排序	328
13.3	我要我的数据——筛选数据	331
13.3.1	实现自动筛选	331
13.3.2	实现高级筛选	333
13.4	格式因条件而定	336
13.4.1	新建条件格式	336
13.4.2	标示最大和最小的 N 个值	339
13.4.3	标示大于或小于平均值的数字	343
13.4.4	使用数据条	345
13.4.5	使用色阶标示数据	348
13.4.6	使用图标集标示数据	351
13.5	单元格特殊内容的输入	355
13.5.1	使用公式	355
13.5.2	向单元格中插入超链接	357
13.5.3	向单元格添加批注	359
13.5.4	限制数据的输入	362

第 14 章 工作表也可以是一张画布	366
14.1 在工作表中使用图形	366
14.1.1 图形类型	366
14.1.2 如何添加图形对象	369
14.1.3 如何使用艺术字	374
14.1.4 如何设置图形的样式	377
14.1.5 大小、位置和角度	381
14.2 使用图表	383
14.2.1 如何引用图表	383
14.2.2 如何创建图表	384
14.2.3 对数据系列进行操作	389
14.2.4 设置图表文字格式	393
14.2.5 对图表区进行操作	395
14.2.6 设置绘图区	397
14.2.7 设置坐标轴	399
14.2.8 为图表添加趋势线	401
14.2.9 导出图表	404
14.2.10 转换图表类型	406
第 15 章 综合案例——员工信息管理系统	410
15.1 制作思路	410
15.1.1 功能简介	410
15.1.2 初始思路	411
15.2 案例制作步骤	411
15.2.1 制作信息表 and 主界面	411
15.2.2 实现新增员工功能	415
15.2.3 实现查询	420
15.2.4 实现修改功能	424
15.2.5 对系统进行保护	426
15.2.6 为宏运行指定按钮	428
15.3 系统功能测试	429
15.3.1 用户登录	429
15.3.2 添加新员工	430
15.3.3 查询员工信息	432
15.3.4 显示信息表 and 退出系统	433

第 1 章 迈向 VBA 的第一步从宏开始

Excel 是一款功能强大的电子表格处理软件，其具有强大的数据分析和处理能力。在一般情况下，Excel 已经能够满足大多数日常数据处理工作的需要，但如果借助于 VBA 语言则可以对数据进行更高级的处理，实现数据处理的自动化和高效化，将工作人员从简单而重复的数据处理工作中解放出来。要了解 VBA，可以首先从 Excel 的宏功能开始。

本章知识点

- 录制宏；
- 宏的运行方式；
- 加载宏；
- 使用宏过程中的问题。



1.1 用宏记录操作

宏是一系列存储于 Visual Basic 模块中的命令和函数，它们记录了用户的一段操作，能够在需要时执行以重现用户的操作过程。宏相当于一段记录用户操作的录像，在任何需要的时候可以重现。因此，使用宏是避免大量重复操作、实现操作自动化的好办法。

1.1.1 录制宏

Excel 具有录制宏的能力，录制宏可以通过宏录制器来进行。录制完成的宏，将作为 VBA 程序代码存储在模块的“代码”窗口中。下面以录制一段向指定的单元格区域中填充颜色的宏为例来介绍录制宏的方法。

(1) 启动 Excel 2013 并创建名为“使用宏”的工作簿，打开“开发工具”选项卡，在“代码”组中单击“录制宏”按钮。此时将打开“录制宏”对话框，在对话框的“宏名”文本框中输入宏名称，将插入点光标放置到“快捷键”文本框后按键即可设置启动宏的快捷键，如这里输入字符 k，再按“Ctrl+K”键将执行宏，如图 1-1 所示。

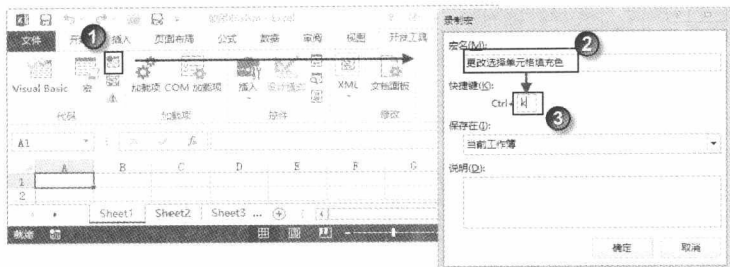


图 1-1 打开“录制宏”对话框

提示：“录制宏”对话框的“保存在”下拉列表用于设置录制的宏的保存位置。当选择“个人宏工作簿”选项时，在其他工作簿中可以使用该宏；如果选择“新工作簿”选项，则录制的宏将保存在新工作簿中；如果选择“当前工作簿”选项，则录制的宏将保存在当前使用的工作簿中，也就是只有当前的工作簿可以使用。

(2) 完成设置后单击“确定”按钮关闭“录制宏”对话框即可开始宏的录制。这里，选择单元格区域，并设置单元格的填充颜色。完成宏录制后，在“开发工具”选项卡的“代码”组中，单击“停止录制”按钮停止宏的录制，如图 1-2 所示。



图 1-2 单击“停止录制”按钮停止宏的录制

使用上述方法录制的宏有一个问题，那就是在使用该宏时，每次都会只对录制宏时选择的单元格进行操作。这是因为在录制宏时，默认使用的是单元格的绝对引用，宏中进行操作的单元格永远是录制时选择的单元格。要解决这个问题实际上很简单，那就是将单元格的引用设置为相对引用。要使用相对引用，在录制宏之前，将“开发工具”选项卡的“代码”组中的“使用相对引用”按钮按下，在进行宏录制时，如果涉及到单元格的引用都将使用相对引用，如图 1-3 所示。

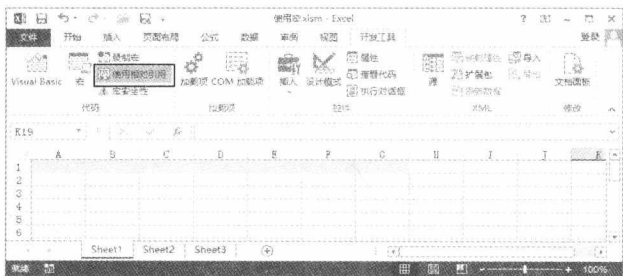


图 1-3 按下“使用相对引用”按钮

1.1.2 宏与VBA

宏的英文名为 Macro，是能够自动执行某种操作的命令集合。宏实际上是由 Office 自动生成的一段 VBA 代码。如在完成上一节宏的录制后，按“Alt+F11”键打开“Visual Basic 编辑器”，可以看到在“工程资源管理器”中会出现一个模块，双击“模块1”选项打开该模块的“代码”窗口，在该窗口中可以看到操作所对应的 VBA 代码，如图 1-4 所示。

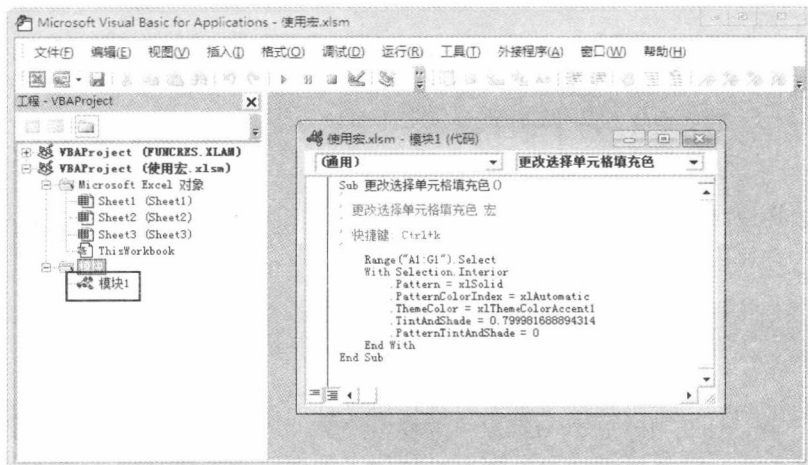


图 1-4 “代码”窗口中的 VBA 代码

录制宏的过程并不能算是编写程序的过程，但是录制宏能够实现与编程相近的功能，能够让需要多个步骤完成的工作一键完成。录制宏可以按照操作顺序如实地记录所有用户操作，所以宏的执行总是产生与录制时的操作完全一致的结果。使用宏的最大优势就在于其使用渐变，录制宏只需要基本的操作，不需要顾及代码的含义、语法和编程思路。只要记录操作，宏代码就自动生成了。

但是使用宏也有其缺点：首先，并非所有的操作都可以通过录制宏来进行，也就是说宏在全面性方法存在着不足；其次，录制宏时无论多么小心，在录制时都有可能产生一些冗余代码，从而降低宏的执行效率；另外，录制宏只能记录操作，生成的宏代码不能进行判断，这也大大限制了宏代码的适用范围；最后，录制宏缺乏灵活性，如录制时选择的单元格或对象，都是以固定模拟过程录制，一旦在操作时发生改变，宏就会产生错误。

因此，录制宏进行操作虽然方便快捷，但其固有的局限性决定了如果需要最大范围扩展 Excel 的功能，最好的方法就是使用 VBA 代码。编写 VBA 程序，无论是在灵活性、执行效率，还是在全面性方面都大大超越了宏。

VBA（即 Visual Basic Application）是针对 Office 软件的一种程序语言，其能够帮助用户从繁杂的重复操作中解放出来，能够提高数据处理的效率，同时对 Excel 的功能进行扩展。实际上，VBA 是一种所见即所得的编程代码，这使得它的学习和使用相对于其他语言要简单。实际上，几乎所有的 VBA 程序员开始都是通过录制宏来学习 VBA 的，这是学习 VBA 的一种速成捷径。甚至对于很多已经成为高手的 VBA 程序员同样对录制宏乐此不疲，因为录制