

<<Excel 2007 VBA与宏完全剖析>>

图书基本信息

书名：<<Excel 2007 VBA与宏完全剖析>>

13位ISBN编号：9787115178589

10位ISBN编号：7115178585

出版时间：2008-7

出版时间：人民邮电出版社

作者：（美）杰莱（Jelen,B.），（美）斯太德（Syrstad,T.）著，郭兵英译

页数：520

字数：783000

译者：郭兵英

版权说明：本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问：<http://www.tushu007.com>

<<Excel 2007 VBA与宏完全剖析>>

内容概要

本书介绍了大量经过实践检验的Excel技巧，帮助读者迅速提高使用VBA开发Excel宏的技能，从而避免大量手工汇总数据和创建报表，节省用户宝贵的时间。

全书包含27章，详细介绍了以下内容：Excel宏录制器和VBA语法，引用区域，使用用户定义函数，循环和流程控制，RIC1公式，使用VBA自动控制Excel 2007新增功能，事件编程，使用用户窗体，创建图表，实现高级筛选，创建数据透视表，数据可视化和条件格式，检索Web数据，使用数组，读写文本文件，创建自定义对象和集合，使用WindowsAPI，处理错误，创建自定义选项卡和加载项。

本书语言简明清晰，内容实用，实例丰富，适合熟悉Excel并想自动完成日常工作或为他人创建Excel应用程序的人员阅读。

<<Excel 2007 VBA与宏完全剖析>>

书籍目录

- 第1章 使用VBA发挥Excel的威力 1.1 Excel的威力 1.2 进入壁垒 1.3 宏录制器不管用 1.3.1 Visual Basic并不像BASIC 1.3.2 VBA非常容易上手 1.3.3 Excel VBA值得学习 1.4 熟悉“开发工具”选项卡 1.5 宏安全性 1.5.1 添加受信任位置 1.5.2 使用宏设置启用不在受信任位置的工作簿中的宏 1.5.3 使用设置“禁用所有宏，并发出通知” 1.6 录制和存储宏 1.7 运行宏 1.7.1 创建宏按钮 1.7.2 将宏关联到窗体控件、文本框或形状 1.8 使用Excel 2007新增的文件类型 1.9 了解Visual Basic编辑器 1.9.1 Visual Basic编辑器设置 1.9.2 工程资源管理器 1.9.3 属性窗口 1.10 了解宏录制器的缺点 1.10.1 准备录制宏 1.10.2 录制宏 1.10.3 在编程窗口中查看代码 1.10.4 其他时间运行该宏时得到意外的结果 1.10.5 一种可能的解决方案：在录制时使用相对引用
- 第2章 VBA简介 2.1 VBA是一种面向对象语言 2.2 VBA语言的组成部分 2.3 VBA学起来并不难 2.3.1 VBA帮助文件-使用F1键获取帮助 2.3.2 使用帮助主题 2.4 查看录制的宏代码——使用Visual Basic编辑器和帮助 2.4.1 可选参数 2.4.2 预定义常量 2.4.3 可返回对象的属性 2.5 使用调试工具帮助理解录制的代码 2.5.1 步进执行代码 2.5.2 另一个调试选项——断点 2.5.3 在代码中向前或向后移动 2.5.4 不逐句运行每行代码 2.5.5 在逐句执行代码时进行查询 2.5.6 使用监视设置断点 2.5.7 监视对象 2.6 所有对象、方法和属性的终极参考信息 2.7 整理代码的5项技巧 2.7.1 技巧1：不要选择任何单元格 2.7.2 技巧2：从区域底部开始查找最后一行 2.7.3 技巧3：使用变量避免硬编码行和公式 2.7.4 技巧4：在一条语句中进行复制和粘贴 2.7.5 技巧5：使用WithEnd With对相同的单元格或区域执行多项操作 2.8 综合应用——修复录制的代码
- 第3章 引用区域 3.1 Range对象 3.2 使用选择的左上角和右下角指定区域 3.3 命名区域 3.4 引用区域的快捷方式 3.5 引用其他工作表中的区域 3.6 引用相对于其他区域的区域 3.7 使用Cells属性选择区域 3.8 使用Offset属性引用区域 3.9 使用Resize属性修改区域的大小 3.10 使用Columns和Rows属性指定区域 3.11 使用Union方法合并多个区域 3.12 使用Intersect方法根据重叠区域创建新区域 3.13 使用IsEmpty函数检查单元格是否为空 3.14 使用CurrentRegion属性快速选择数据区域 3.15 使用区域集合返回非连续区域 3.16 引用数据表
- 第4章 用户定义的函数 4.1 创建用户定义的函数 4.2 自定义函数示例 4.3 共享UDF 4.4 很有用的自定义函数 4.4.1 获取工作簿的名称 4.4.2 获取当前工作簿的名称和文件路径 4.4.3 检查工作簿是否打开 4.4.4 检查工作表是否包含在打开的工作簿中 4.4.5 统计文件夹中的工作簿数量 4.4.6 获取UserID 4.4.7 检索最后一次保存的日期和时间 4.4.8 获取固定不变的日期和时间 4.4.9 验证电子邮件地址 4.4.10 根据内部颜色对单元格求和 4.4.11 统计非重复值的数量 4.4.12 删除区域中重复的值 4.4.13 在区域中查找第一个非空单元格 4.4.14 替换多个字符 4.4.15 从文本中获取数字 4.4.16 将星期编号转换为日期 4.4.17 从用分隔符分隔的字符串中提取值 4.4.18 排序并拼接 4.4.19 对数字和字母进行排序 4.4.20 在文本中搜索字符串 4.4.21 颠倒单元格内容的顺序 4.4.22 多个最大值 4.4.23 返回链接的地址 4.4.24 返回单元格地址的列字母 4.4.25 静态随机 4.4.26 将Select Case用于工作表
- 第5章 循环和流程控制 5.1 ForNext循环 5.1.1 在For语句中使用变量 5.1.2 ForNext循环的变体 5.1.3 在满足特定条件时提早退出循环 5.1.4 嵌套循环 5.2 Do循环 5.2.1 在Do循环中使用While或Until子句 5.2.2 While...Wend循环 5.3 VBA循环：For Each 5.3.1 对象变量 5.3.2 遍历文件夹中的所有文件 5.4 流程控制：使用If...The Else和Select Case 5.4.1 基本流程控制：If...The Else 5.4.2 条件 5.4.3 If...The ElseEnd If 5.4.4 使用IfElse IfEnd If检测多个条件 5.4.5 使用Select CaseEnd Select检测多个条件 5.4.6 在Case语句中使用复杂表达式 5.4.7 嵌套If语句
- 第6章 R1C1公式 6.1 引用单元格：A1和R1C1引用样式之比较 6.2 切换到R1C1引用样式 6.3 Excel公式创造的奇迹 6.3.1 输入公式一次并复制多次 6.3.2 速并不神奇 6.3.3 在VBA中采用A1与R1C1样式之比较 6.4 R1C1引用样式简介 6.4.1 使用R1C1相对引用 6.4.2 使用R1C1绝对引用 6.4.3 使用R1C1混合引用 6.4.4 使用R1C1引用样式整行或整列 6.4.5 使用单个R1C1公式替换多个A1公式 6.4.6 记住列字母对应的列号 6.5 条件格式必须使用R1C1引用 6.5.1 通过用户界面设置条件格式 6.5.2

<<Excel 2007 VBA与宏完全剖析>>

使用VBA设置条件格式 6.5.3 找出G列中最大的值 6.6 数组公式必须是R1C1公式第7章
 Excel 2007新增和修改过的功能 7.1 如果Excel用户界面不同,则VBA也不同 7.1.1 选项卡
 7.1.2 图表 7.1.3 数据透视表 7.1.4 条件格式 7.1.5 数据表 7.1.6 排序 7.1.7
 SmartArt 7.2 宏录制器不记录在以前的版本中记录的操作 7.3 学习新的对象和方法 7.4 兼容
 模式 7.4.1 Version 7.4.2 Excel8CompatibilityMode第8章 使用VBA创建和操作名称 8.1
 Excel名称 8.2 全局名称和局部名称 8.3 添加名称 8.4 删除名称 8.5 添加备注 8.6 名称
 类型 8.6.1 公式 8.6.2 字符串 8.6.3 数字 8.6.4 数据表 8.6.5 让名称指向数组
 8.6.6 保留名称 8.7 隐藏名称 8.8 确定名称是否存在第9章 事件编程 9.1 事件级别 9.2
 使用事件 9.2.1 事件参数 9.2.2 启用事件 9.3 工作簿事件 9.3.1 Workbook Activate ()
 9.3.2 Workbook Deactivate () 9.3.3 Workbook_Open () 9.3.4 Workbook BeforeSave
 (ByVal SaveAsUI As Boolean , Cancel As Boolean) 9.3.5 Workbook_BeforePrint (Cancel As Boolean
) 9.3.6 Workbook BeforeClose (Cancel As Boolean) 9.3.7 Workbook_NewSheet (ByVal Sh As
 Object) 9.3.8 Workbook_WindowResize (ByVal Wn As Window) 9.3.9 Workbook
 _WindowActivate (ByVal Wn As Window) 9.3.10 Workbook_WindowDeactivate (ByVal Wn As
 Window) 9.3.11 Workbook_AddInInstall () 9.3.12 Workbook_AddInUninstall 9.3.13
 Workbook_SheetActivate (ByVal Sh As Object) 9.3.14 Workbook_SheetBeforeDoubleClick (ByVal
 Sh As Object, ByVal Target As Range, Cancel As Boolean) 9.3.15 Workbook_SheetBeforeRightClick
 (ByVal Sh As Object, ByVal Target As Range, Cancel As Boolean) 9.3.16 Workbook_SheetCalculate
 (ByVal Sh As Object) 9.3.17 Workbook_SheetChange (ByVal Sh As Object, ByVal Target As Range)
 9.3.18 Workbook_Sync (ByVal SyncEventType As Office.MsoSyncEventType) 9.3.19
 Workbook_SheetDeactivate (ByVal Sh As Object) 9.3.20 Workbook_SheetFollowHyperlink (ByVal
 Sh As Object, ByVal Target As Hyperlink) 9.3.21 Workbook_SheetSelectionChange (ByVal Sh As
 Object, ByVal Target As Range) 9.3.22 Workbook PivotTableCloseConnection (ByVal Target As
 PivotTable) 9.3.23 Workbook_PivotTableOpenConnection (ByVal Target As PivotTable) 9.3.24
 Workbook RowsetComplete (ByVal Description As String, ByVal Sheet As String, ByVal Success As Boolean
) 9.4 工作表事件 9.4.1 Worksheet_Activate () 9.4.2 Worksheet_Deactivate () 9.4.3
 Worksheet BeforeDoubleClick (ByVal Target As Range, Cancel As Boolean) 9.4.4 Worksheet
 BeforeRightClick (ByVal Target As Range, Cancel As Boolean) 9.4.5 Worksheet_Calculate0 9.4.6
 Worksheet Change (ByVal Target As Range) 9.4.7 Worksheet_SelectionChange (ByVal Target As
 Range) 9.4.8 Worksheet_FollowHyperlink (ByVal Target As Hyperlink) 9.5 在单元格中快速输
 入24小时制事件 9.6 图表事件 9.6.1 嵌入图表 9.6.2 Chart_Activate () 9.6.3
 Chart BeforeDoubleClick (ByVal ElementID As Long, ByVal Arg1 As Long, ByVal Arg2 As Long, Cancel As
 Boolean) 9.6.4 Chart BeforeRightClick (Cancel As Boolean) 9.6.5 Chart_Calculate ()
 9.6.6 Chart_Deactivate () 9.6.7 Chart_MouseDown (ByVal Button As Long, ByVal Shift As
 Long, ByVal x As Long, ByVal y As Long) 9.6.8 Chart MouseMove (ByVal Button As Long, ByVal
 Shift As Long, ByVal x As Long, ByVal y As Long) 9.6.9 Chart_MouseUp (ByVal Button As Long,
 ByVal Shift As Long, ByVal x As Long, ByVal y As Long) 9.6.10 Chart Resize () 9.6.11
 Chart_Select (ByVal ElementID As Long , ByVal Arg1 As Long , ByVal Arg2 As Long 9.6.12
 Chart_SeriesChange (ByVal SeriesIndex As Long , ByVal PointIndex As Long) 9.6.13
 Chart_DragOver () 9.6.14 Chart_DragPlot () 9.7 应用程序级事件 9.7.1
 AppEvent_AfterCalculate0 9.7.2 AppEventNewWorkbook (ByVal Wb As Workbook、 9.7.3
 AppEvent_SheetActivate (ByVal Sh As Object) 9.7.4 AppEvent_SheetBeforeDoubleClick (ByVal
 Sh As Object, ByVal Target As Range , Cancel As Boolean) 9.7.5 AppEvent SheetBeforeRightClick
 (ByVal Sh As Object , ByVal Target As Range , Cancel As Boolean) 9.7.6 AppEvent SheetCalculate
 (ByVal Sh As Object) 9.7.7 AppEvent_SheetChange (ByVal Sh As Object , ByVal Target As Range
) 9.7.8 AppEvent_SheetDeactivate (ByVal Sh As Object) 9.7.9
 AppEvent_SheetFollowHyperlink (ByVal Sh As Object, ByVal Target As Hyperlink) 9.7.10

<<Excel 2007 VBA与宏完全剖析>>

AppEvent_SheetSelectionChange (ByVal ShAs Object.ByVal Target As Range)	9.7.11
AppEvent_WindowActivate (ByVal Wb As Workbook , ByVal Wn As Window)	9.7.12
AppEvent_WindowDeactivate (ByVal Wb As Workbook , ByVal Wn As Window)	9.7.13
AppEvent_WindowResize (ByVal Wb As Workbook.ByVal Wn As Window)	9.7.14
AppEvent_WorkbookActivate (ByVal WbAs Workbook)	9.7.15
AppEvent WorkbookAddinInstall (ByVal WbAs Workbook)	9.7.16
AppEvent WorkbookAddinUninstall (ByVal WbAs Workbook, 9.7.17	
AppEvent WorkbookBeforeClose (ByVal Wb As Workbook.Cancel As Boolean ByVal Target As PivotTable)	9.7.18
AppEvent WorkbookBeforePrint (ByVal Wb As Workbook.Cancel As Boolean)	9.7.19
AppEvent WorkbookBeforeSave (ByVal Wb As Workbook.Byval SaveAsUI As Boolean , Cancel As Boolean)	9.7.20
AppEvent WorkbookNewSheet (ByVal Wb As Workbook.ByVal Sh As Object	9.7.21
AppEvent WorkbookOpen (ByVal Wb As Workbook)	9.7.22
AppEvent WorkbookPivotTableCloseConnection (ByVal Wb As Workbook.ByVal Target As PivotTable)	9.7.23
AppEvent WorkbookPivotTableOpenConnection (ByVal Wb As Workbook.	9.7.24
AppEvent WorkbookRowsetComplete (ByVal Wb As Workbook, ByVal Description As String, ByVal Sheet As String, ByVal Success As Boolean)	9.7.25
AppEvent WorkbookSync (ByVal Wb As Workbook, ByVal SyncEventType As Office.MsoSyncEventType)	

第10章 用户窗体简介第11章 创建图表第12章 使用统计函数第13章 使用VBA创建数据透视表第14章 展示Excel的威力第15章 数据可视化和条件格式第16章 读写Web第17章 Excel 2007中的XML第18章 自动控制Word第19章 数组第20章 处理文本文件第21章 将Access用作后端以及改善多用户数据访问第22章 创建类、记录和集合第23章 高级用户窗体技术第24章 Windows应用程序编程接口第25章 处理错误第26章 创建自定义选项卡方便用户运行宏第27章 创建加载项

章节摘录

第1章 使用VBA发挥Excel的威力 1.1 Excel的威力 结合使用Visual Basic for Applications (VBA) 和Microsoft Excel可能是最强大的工具, 该工具就位于5亿Microsoft Office用户的桌面上, 但大多数用户并不知道如何发挥Excel VBA的威力。

通过使用VBA, 可更快地在Excel中完成各种任务。

如果读者定期使用Excel生成一系列的月度图表, 可让VBA帮助完成这种任务, 这只需几秒钟。

1.2 进入壁垒 要成功地学习VBA编程, 将遇到两个壁垒: 首先, Excel的宏录制器存在缺陷, 无法生成可作为模型使用的代码; 其次, 对于学习过编程语言(如Basic)的用户来说, VBA的语法难以理解。

1.3 宏录制器不管用 Microsoft在20世纪90年代开始占领电子表格市场。虽然他们在建立功能强大的电子表格程序让Lotus 1-2-3用户可轻松过渡方面取得了巨大成功, 但其宏语言截然不同, 精通录制Louts 1-2-3宏的用户在尝试录制Excel宏时将遇到困难。

虽然Microsoft VBA编程语言的功能远远超过了Lotus 1-2-3的宏语言, 但一个重要的缺陷是宏录制品不管用。

<<Excel 2007 VBA与宏完全剖析>>

编辑推荐

“在当今这个信息浩如烟海而时间弥足珍贵的时代、只有以简洁的方式快速获悉汇总数据，才能够让公司脱颖而出成为行业领头羊本书介绍的技巧将让您梦想成真”——Jerry Kohl, Brighton Collectibles公司顾问 如果您是Excel专家，却发现宏录制器不起作用，且录制的代码让您一头雾水，则本书正是为您编写的。

今天录制的宏可能在明天不起作用，也就是说，录制的宏可能能够处理包含14条记录的数据集，但无法处理包含12条或16条记录的数据集。

这是宏录制器存在的常见问题，它们是导致众多Excel大师厌恶编写宏的罪魁祸首。

本书讨论宏录制器不起作用的原因。

阐述如何对录制的代码进行修改，使其能够处理任何数据集。

本书假设读者精通Excel，但不要求读者有任何编程经验。

本书全面介绍了使用Excel VBA自动生成报表和设计应用程序所需的技能，适合要自动生成报表或为他人设计功能齐全的应用程序的读者阅读。

- 学习VBA语法；
- 自动控制功能强大的Excel工具：数据透视表、图表和高级筛选；
- 自动完成重复的任务；
- 使用自定义对话框创建Excel应用程序；
- 快速生成数百个Excel个报表；
- 了解Excel 2007中的变化对VBA宏的影响。

学习经过实践检验的技巧，迅速提高使用VBA开发Excel宏的技能。

Bill Jelen，就是著名的MrExcel！

他是业界领先的Excel网站MrExcel.com的负责人和Microsoft Excel的MVP。

他在一家快速成长的计算机公司担任了12年的财务分析师，在此期间其Excel技能变得出神入化。

只需凭借一个电子表格，他就能够在最短的时间内将数千行的交易数据变成有意义的摘要报告。

Bill Jelen是一名卓有成就的Excel图书作者。

还是加拿大电视节目的嘉宾。

在世界各地的会计俱乐部常常能够见到Bill的身影，那是他正在举办幽默而内容丰富的Excel讲座。

他的网站每年的访问量高达1200万次。

Lacy Svrstad 是Excel和Access咨询师。

作为MrExcel.com顾问团队的一员。

她帮助世界各地的客户开发VBA解决方案。

她对以前为开发应用程序而学习VBA的艰辛仍记忆犹新。

版权说明

本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问:<http://www.tushu007.com>