

专业级 Excel VBA 培训课程：用 VBA 进行 Excel 二次开发

课程简介

本课程系统讲述了 Excel VBA 开发的基本概念、VBA 语言、Excel 对象模型、应用 Excel 和 VBA 进行数据处理、Excel 图表的 VBA 编程、用户窗体和 Excel 工作表界面设计、开发效率、互操作、API 调用等高级话题；并结合 3 个实际案例，介绍了如何在工作实际中使用 VBA 和 Excel 提高工作效率，设计 Excel VBA 应用程序。

培训对象

本课程适用于想通过 Excel VBA 来提高工作效率，使用 Excel 和 VBA 作为开发和数据处理平台的人。

培训大纲

第 1 章 Excel 和 VBA 简介

- 1.1 关于 Excel 和 VBA
- 1.2 Excel 作为数据处理平台
- 1.3 Excel 作为开发平台
- 1.4 宏、加载宏和 VBA

第 1 部分 入门篇

第 2 章 Excel 与宏

- 2.1 宏
- 2.2 录制宏
 - 2.2.1 宏的录制步骤
 - 2.2.2 录制宏的一个小例子
- 2.3 运行宏
 - 2.3.1 运行录制的宏
 - 2.3.2 将宏加入菜单或工具栏
- 2.4 查看和编辑宏
- 2.5 加载宏

第 3 章 VBA 初步

- 3.1 VBA 简介
- 3.2 VBA 快速入门
 - 3.2.1 VBA 开发环境
 - 3.2.2 VBA 工程及其组成
 - 3.2.3 编写代码
 - 3.2.4 模块、过程和函数

- 3.2.5 创建过程和函数
- 3.2.6 过程调用
- 3.2.7 变量
- 3.2.8 程序流程
- 3.2.9 条件语句
- 3.2.10 循环语句
- 3.2.11 数组
- 3.2.12 善用工具及其他
- 3.3 应用 VBA 操作 Excel
 - 3.3.1 Excel 对象模型
 - 3.3.2 Excel 对象模型中的常用对象
- 3.4 应用实例
 - 3.4.1 使用 VBA 合并列
 - 3.4.2 自动隐藏或显示表格中无数据的行
 - 3.4.3 使用 VBA 操作工作表单元格
 - 3.4.4 查找工作表的第一个空行
 - 3.4.5 改变 Excel 界面的标题
 - 3.4.6 隔行格式化工作表
 - 3.4.7 批量修改数据
- 第 4 章 应用 VBA 自定义函数
 - 4.1 公式和函数
 - 4.1.1 公式及其组成
 - 4.1.2 引用
 - 4.1.3 名称和常量
 - 4.1.4 运算符和运算次序
 - 4.2 Excel 工作表函数
 - 4.2.1 常用函数
 - 4.2.2 统计函数
 - 4.2.3 数组公式
 - 4.3 自定义函数
 - 4.3.1 创建自定义函数
 - 4.3.2 创建数组公式
 - 4.4 应用实例
 - 4.4.1 成绩统计辅助函数
 - 4.4.2 身份证号码处理函数

第2部分 进阶篇

第5章 VBA 语言

5.1 VBA 程序的组成

5.1.1 模块

5.1.2 过程

5.1.3 函数

5.1.4 过程和函数调用

5.1.5 注释

5.2 数据类型

5.2.1 常量和变量

5.2.2 数据类型

5.2.3 运算符

5.2.4 数组

5.2.5 自定义数据类型

5.2.6 枚举类型

5.2.7 变量的作用域（生存周期）

5.2.8 字符串

5.2.9 日期和时间

5.3 控制程序流程

5.3.1 条件语句

5.3.2 循环语句

5.3.3 With 语句

5.3.4 Exit 语句

5.4 简单的用户交互

5.5 用户窗体

5.5.1 设计用户窗体

5.5.2 事件驱动

5.5.3 使用控件

5.6 调试 VBA 代码

5.6.1 错误的类型

5.6.2 使用 Debug 对象

5.6.3 VBA 的调试工具

5.7 错误处理

5.7.1 设置错误捕获

5.7.2 编写错误处理实用程序

5.7.3 提供从错误处理程序跳出的出口

5.7.4 错误处理的简单示例

5.8 类模块和面向对象

5.8.1 面向对象开发

5.8.2 对象变量和对象

5.8.3 创建类模块

5.9 COM 对象的使用

5.10 集合对象

第 6 章 Excel 的对象模型

6.1 Excel 对象模型简介

6.2 Application 对象

6.2.1 控制 Excel 状态和显示

6.2.2 返回对象

6.2.3 执行操作

6.2.4 Window 对象和 Windows 集合

6.2.5 Application 事件

6.3 Workbook 对象

6.3.1 Workbooks 集合

6.3.2 Workbook 的属性

6.3.3 Sheets 集合

6.3.4 Workbook 的方法

6.3.5 Workbook 的事件

6.4 Worksheet 对象

6.5 Range 对象

6.5.1 返回或获得 Range 对象

6.5.2 Range 对象的常用属性和方法

第 3 部分 专题篇

第 7 章 数据处理

7.1 概述

7.2 Excel 数据处理的方式和流程

7.2.1 方式和流程

7.2.2 “表格驱动”的数据处理

7.2.3 基于“过程”方法的数据处理

7.2.4 基于“面向对象”方法的数据处理

7.3 操作数据文件

7.3.1 使用 Excel 对象操作数据文件

7.3.2 使用 VBA 语句操作文件

7.3.3 FileSystemObject 对象模型

7.3.4 Excel 的数据导入功能

7.4 操作数据库

7.4.1 使用 Excel 工具导入数据库的数据

7.4.2 使用 VBA 来操作查询表

7.4.3 使用 ADO 操作数据库

7.5 操作 Web 数据

7.6 操作数据

7.6.1 工作表数据引用

7.6.2 操作文本

7.6.3 操作数值

7.6.4 操作日期和时间

7.6.5 Excel 工作表函数

7.6.6 数据类型的判断和转换

7.7 应用实例

7.7.1 格式化数据

7.7.2 在 Excel 中使用 VBA 来筛选数据

7.7.3 批量重命名文件

7.7.4 文本信息处理实例

第 8 章 Excel 图表

8.1 Excel 图表及其对象

8.2 创建 Excel 图表

8.2.1 创建图表工作表

8.2.2 创建嵌入式图表

8.2.3 通过录制宏来创建图表

8.3 操作 Excel 图表

8.3.1 激活图表

8.3.2 修改图表的数据系列

8.3.3 图表的格式

8.3.4 图表的输出

8.4 Excel 图表的事件

8.5 Excel 图表的对象模型

8.6 应用实例：Excel 下三角图解的绘制

8.6.1 问题描述

8.6.2 解决思路

8.6.3 使用宏代码绘制三角图

第9章 界面设计

第4部分 实例篇

第10章 应用实例：批量数据导入工具

第11章 应用实例：基于 Excel 的查询工具

第12章 应用实例：GCDPlot

第5部分 高级篇

第13章 高级话题

13.1 调用 Windows

13.1.1 在 VBA 中使用 API

13.1.2 简单的 API 调用

13.1.3 封装 API 调用为函数

13.1.4 封装 API 调用为类模块

13.2 自动化其他 Office 组件

13.2.1 启动其他 Office 组件

13.2.2 与其他 Office 组件交互

13.3 使用 Office 助手

13.4 Excel VBA 程序的类型和部署

13.4.1 Excel VBA 程序的类型

13.4.2 加载宏和一般电子表格程序的优缺点

13.4.3 部署

13.5 VBA 程序的安全性和保护

13.6 通过其他程序自动化 Excel

13.6.1 创建 Excel 对象

13.6.2 Excel 自动化中的事件

13.6.3 使用 Excel 完成业务逻辑

13.7 Excel 数据导入导出的几种方式

13.7.1 使用自动化传输数据

13.7.2 使用 ADO 操作 Excel 数据

13.7.3 使用第三方类库

13.8 关于 Excel 工程的引用

13.9 提高效率的一些建议

13.9.1 尽量使用 Excel 的工作表函数

13.9.2 尽量减少使用对象引用

13.9.3 高效使用 Range 对象

13.9.4 减少对象的激活和选择

13.9.5 关闭屏幕更新

13.9.6 提高关键代码的效率

13.9.7 代码执行时间的测算