

## VBA Programming: A Comprehensive Hands-On Introduction - 4 Days

### *Course 209 Overview*

#### **You Will Learn How To**

- Develop customized solutions using Visual Basic for Applications (VBA)
- Apply the object models to program effectively in the Microsoft Office System
- Take full advantage of extensive built-in VBA functionality
- Exploit ActiveX controls to create functional and familiar user interfaces
- Troubleshoot code using VBA's debugging tools
- Secure your code and protect your users from macro viruses

#### **Course Benefits**

Effective business solutions do not come straight out of the box. With VBA, you can write code to enhance the off-the-shelf capabilities of Office and other products that support VBA. Using VBA, you can develop custom-built solutions. In this course, you learn the fundamentals of programming and VBA from the ground up. You gain the skills you need to automate repetitive tasks, build user interfaces and increase performance.

#### **Who Should Attend**

This course is valuable for anyone who wants to automate tasks through VBA programming. Some experience using the products in the Microsoft Office System is assumed. No previous programming experience is required.

#### **Hands-On Training**

Extensive hands-on exercises provide you with practical experience automating Office applications, such as Word, Excel and Outlook, with VBA programming. Exercises include:

- Building user interfaces with standard and custom controls
- Adding powerful subtotaling to an Excel workbook
- Creating form letters and Outlook mail with customized userforms
- Protecting users with digital signatures
- Retrieving Access and SQL Server data for use in Word and Excel
- Integrating the powerful features of Word and Excel

# VBA Programming: A Comprehensive Hands-On Introduction - 4 Days

## Course 209 Outline

### Introduction

#### The advantages of VBA

- An easy-to-use and versatile language
- Integrated with off-the-shelf products
- Minimizes development time and costs

#### Macros: the first step

- Recording and executing macros
- Limitations
- Enhancing with VBA code

### Getting to Know the VBA Development Environment

#### The code design tools

- Project Explorer
- Help system
- Code Window
- IntelliSense
- Properties Window
- Object Browser
- Block commenting and uncommenting

#### The debugging tools

- Code stepping
- Immediate Window
- Breakpoints

### Writing VBA Code

#### Event-driven programming

- The role of events in Windows
- How events are triggered
- Responding to events through event procedures

#### Putting objects to work

- Properties
- Methods
- Events
- Collections
- Using With...End With

#### Storing data in variables

- How and when to declare variables
- Selecting data types
- Fixed and dynamic arrays
- Constants
- The scope and lifetime of variables

#### Conditional branching

- If...Then...Else
- Select...Case

#### Looping through code

- Do...Loop, While and Until
- For...Next and For Each...Next

#### Building procedures

- Creating Sub and Function procedures
- Calling procedures
- Passing arguments to procedures

### Empowering VBA through the Application Object Models

#### Understanding object model structures

- Exploring the hierarchy using the Object Browser and Help system
- Referencing specific objects

#### Writing application-specific procedures

- Key objects within Excel, Word and Outlook
- Creating templates and add-ins

#### Exploiting the power of Automation

- Controlling one Office application while working in another application
- Accessing data in databases using ActiveX Data Objects

### Building Intuitive User Interfaces

#### Using intrinsic dialogs

- Communicating with the user through the message box
- Gathering user information with the input box
- Utilizing the FileDialog object and Dialogs collections

#### Creating customized dialogs with UserForm objects

- Command buttons
- List and Combo boxes
- Option buttons
- Check boxes
- Labels
- Text boxes
- Frames
- Adding more functionality with advanced ActiveX controls

#### Modifying menus and toolbars

- Simplifying user interaction with the CommandBar object
- Adding and removing CommandBarControl objects

### Providing a Safe and Secure Environment

#### Handling runtime errors

- The On Error GoTo structure
- Classifying errors with the Err object
- Continuing execution with Resume, Resume Next or Resume label

#### Implementing security

- The dangers of macro viruses
- Macro security levels in Office
- Applying digital signatures to macros
- Password protecting your VBA code