

**Excel 2019 VBA**

**Course Number:** VBA-134  
**Duration:** 3 days

**Overview**

Accelebrate's Excel 2019 VBA training teaches attendees how to automate Excel 2019 using Visual Basic for Applications.

**Important Note:** We also offer a [4-day](file:////training/vba-excel-2019-extended) course that course includes a day of time spent working on your actual VBA projects, existing and under development.

**Prerequisites**

All attendees must have prior knowledge of Excel 2019. Prior scripting or programming knowledge is very helpful but not required.

**Materials**

All Excel VBA training attendees receive comprehensive courseware.

**Software Needed on Each Student PC**

* A full installation of Microsoft Office 2019 (or at least Excel and Access), including Visual Basic support and Visual Basic Help.
* If you have purchased this course, please contact us for detailed setup instructions.

**Objectives**

All attendees will:

* Use the main features of the VBA Editor window and learn core VBA concepts.
* Build sub procedures and user-defined functions.
* Learn the Excel object model and write code to control Excel objects.
* Create and use variables.
* Use a wide array of standard programming techniques.
* Create a user interface (a custom form complete with a variety of controls and code to drive the user form).
* Create PivotTables programmatically.
* Learn and use multiple troubleshooting and debugging features.
* Incorporate error handlers to deal with unforeseen issues.

**Outline**

* Getting Started
  + Introducing Visual Basic for Applications
  + Displaying the Developer Tab in the Ribbon
  + Recording a Macro
  + Saving a Macro-Enabled Workbook
  + Running a Macro
  + Editing a Macro in the Visual Basic Editor
  + Understanding the Development Environment
  + Using Visual Basic Help
  + Closing the Visual Basic Editor
  + Understanding Macro Security
* Working with Procedures and Functions
  + Understanding Modules
  + Creating a Standard Module
  + Understanding Procedures
  + Creating a Sub Procedure
  + Calling Procedures
  + Using the Immediate Window to Call Procedures
  + Creating a Function Procedure
  + Naming Procedures
  + Working with the Code Editor
* Understanding Objects
  + Understanding Objects
  + Navigating the Excel Object Hierarchy
  + Understanding Collections
  + Using the Object Browser
  + Working with Properties
  + Using the With Statement
  + Working with Methods
  + Creating an Event Procedure
* Using Expressions, Variables, and Intrinsic Functions
  + Understanding Expressions and Statements
  + Declaring Variables
  + Understanding Data Types
  + Working with Variable Scope
  + Using Intrinsic Functions
  + Understanding Constants
  + Using Intrinsic Constants
  + Using Message Boxes
  + Using Input Boxes
  + Declaring and Using Object Variables
* Controlling Program Execution
  + Understanding Control-of-Flow Structures
  + Working with Boolean Expressions
  + Using the If...End If Decision Structures
  + Using the Select Case...End Select Structure
  + Using the Do...Loop Structure
  + Using the For...To...Next Structure
  + Using the For Each...Next Structure
  + Guidelines for use of Control-of-Flow Structures
* Working with Forms and Controls
  + Understanding UserForms
  + Using the Toolbox
  + Working with UserForm Properties, Events, and Methods
  + Understanding Controls
  + Setting Control Properties in the Properties Window
  + Working with the Label Control
  + Working with the Text Box Control
  + Working with the Command Button Control
  + Working with the Combo Box Control
  + Working with the Frame Control
  + Working with Option Button Controls
  + Working with Control Appearance
  + Setting the Tab Order
  + Populating a Control
  + Adding Code to Controls
  + Launching a Form in Code
* Working with the PivotTable Object
  + Understanding PivotTables
  + Creating a PivotTable Using Worksheet Data
  + Working with the PivotTable Objects
  + Working with the PivotFields Collection
  + Assigning a Macro to the Quick Access Toolbar
* Debugging Code
  + Understanding Errors
  + Using Debugging Tools
  + Setting Breakpoints
  + Stepping through Code
  + Using Break Mode during Run mode
  + Determining the Value of Expressions
* Handling Errors
  + Understanding Error Handling
  + Understanding VBA's Error Trapping Options
  + Trapping Errors with the On Error Statement
  + Understanding the Err Object
  + Writing an Error-Handling Routine
  + Working with Inline Error Handling
* Conclusion