MOC 50432 Excel 2007 Programming Using Visual Basic for Applications

Course Summary

Description

This course provides students with the knowledge and skills to use Microsoft Visual Basic for Applications to control Microsoft Excel 2007 programmatically. The course serves as an introduction to programming using the Visual Basic programming language.

Objectives

At the end of this course, students will be able to:

- Record and edit macros.
- Use the Visual Basic Editor.
- Create and call sub and function procedures.
- Understand objects, properties, methods, and events.
- Use the Object Browser.
- Work with variables and understand data types.
- Use intrinsic functions.
- Work with control-of-flow structures.
- Design UserForms and work with controls.
- Control PivotTables programmatically.
- Use debugging tools.
- Add error handling to code.

Topics

- Getting Started
- Working with Procedures and Functions
- Understanding Objects
- Using Expressions, Variables, and Intrinsic Functions
- Controlling Program Execution
- Working with Forms and Controls
- Working with the PivotTable Object
- Debugging Code

Audience

This course is intended for experienced Excel users who are interested in creating code to automate tasks in Excel.

Prerequisites

Before attending this course, students must have:

- An intermediate level of experience using Microsoft Excel 2007. Familiarity with PivotTables is helpful, but not required. No prior programming experience is necessary.

Duration

Three days

Due to the nature of this material, this document refers to numerous hardware and software products by their trade names. References to other companies and their products are for informational purposes only, and all trademarks are the properties of their respective companies. It is not the intent of ProTech Professional Technical Services, Inc. to use any of these names generically

MOC 50432 Excel 2007 Programming Using Visual Basic for Applications

Course Outline

I. Getting Started

This module explains how to record and run macros. It also describes the Visual Basic Editor and how to use Visual Basic Help.

A. Introducing Visual Basic for Applications
B. Displaying the Developer Tab in the Ribbon
C. Recording a Macro
D. Saving a Macro-Enabled Workbook
E. Running a Macro
F. Editing a Macro in the Visual Basic Editor
G. Understanding the Development Environment
H. Using Visual Basic Help
I. Closing the Visual Basic Editor
J. Understanding Macro Security

Lab: Getting Started

- 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
- Creating a Trusted Location (Optional)
- Introduction to the Course Exercises

After completing this module, students will be able to:

- Record and run a macro.
- Edit a macro in the Visual Basic Editor.
- Use Visual Basic Help.

II. Working with Procedures and Functions

This module explains how to create modules, sub procedures, and function procedures. It also describes how to call procedures and work with the Code Editor.

A. Understanding Modules
B. Creating a Standard Module
C. Understanding Procedures
D. Creating a Sub Procedure
E. Calling Procedures
F. Using the Immediate Window to Call Procedures
G. Creating a Function Procedure
H. Naming Procedures
I. Working with the Code Editor

Lab: Working with Procedures and Functions

- Creating a Standard Module
- Creating a Sub Procedure
- Using the Immediate Window to Call Procedures
- Creating a Function Procedure
- Calling a User-Defined Function from Excel
- Working with the Code Editor

After completing this module, students will be able to:

- Create modules, sub procedures, and function procedures.
- Write code to call a procedure.
- Call a procedure from the Immediate Window.
- Work with the Code Editor.

III. Understanding Objects

This module explains Excel objects and object properties, methods, and events. It details how to work with objects in code and how to use the Object Browser.

A. Understanding Objects
B. Navigating the Excel Object Hierarchy
C. Understanding Collections
D. Using the Object Browser
E. Working with Properties
F. Using the With Statement
G. Working with Methods
H. Creating an Event Procedure

Lab: Understanding Objects

- Understanding Collections
- Using the Object Browser
- Working with Properties
- Using the With Statement

Due to the nature of this material, this document refers to numerous hardware and software products by their trade names. References to other companies and their products are for informational purposes only, and all trademarks are the properties of their respective companies. It is not the intent of ProTech Professional Technical Services, Inc. to use any of these names generically

MOC 50432 Excel 2007 Programming Using Visual Basic for Applications

Course Outline (cont'd)

- Working with Methods
- Working with Events

After completing this module, students will be able to:

- Refer to Excel objects programmatically.
- Use the Object Browser.
- Write code to read and change properties of an object.
- Use the With End With statement.
- Write code to call a method.
- Create an event procedure.

IV. Using Expressions, Variables, and Intrinsic Functions

This module explains how to name, declare, assign values to, and use variables. It also describes the use of user-defined constants and Visual Basic intrinsic constants. The use of Message Boxes, Input Boxes and object variables is also covered.

A. Understanding Expressions and Statements
B. Declaring Variables
C. Understanding Data Types
D. Working with Variable Scope
E. Using Intrinsic Functions
F. Understanding Constants
G. Using Intrinsic Constants
H. Using Message Boxes
I. Using Input Boxes
J. Declaring and Using Object Variables

Lab: Using Expressions, Variables, and Intrinsic Functions

- Declaring Variables Explicitly
- Displaying the Compile Project Command Button
- Declaring Variables
- Using Intrinsic Functions
- Using Message Boxes
- Using Input Boxes
- Declaring and Using Object Variables

After completing this module, students will be able to:

- Declare variables.
- Assign values to variables.
- Use intrinsic functions.
- Use the MsgBox and InputBox functions.
- Use intrinsic constants.
- Declare and set object variables.

V. Controlling Program Execution

This module explains how to use decision structures to control program flow. It also covers how to use looping structures to repeat a block of code.

A. Understanding Control-of-Flow Structures
B. Working with Boolean Expressions
C. Using the If...End If Decision Structures
D. Using the Select Case...End Select Structure
E. Using the Do...Loop Structure
F. Using the For...To...Next Structure
G. Using the For Each...Next Structure
H. Guidelines for use of Control-of-Flow Structures

Lab: Controlling Program Execution

- Working with the If…Then...End If Structure
- Using the If…Then…Else…End If Structure
- Using the If…Then…ElseIf…End If Structure
- Using the Select Case…End Select Structure
- Using the Do…Loop Structure
- Finishing the GetNewInventory Procedure
- Using the For…To…Next Structure
- Using the For Each…Next Structure

After completing this module, students will be able to:

- Understand Boolean expressions used in program flow structures.
- Use comparison operators and logical operators to build expressions.
- Use **If** decision structures.
- Use the **Select Case** decision structure.
- Use the **Do … Loop** looping structure.
- Use **For … Next** looping structures.

Due to the nature of this material, this document refers to numerous hardware and software products by their trade names. References to other companies and their products are for informational purposes only, and all trademarks are the properties of their respective companies. It is not the intent of ProTech Professional Technical Services, Inc. to use any of these names generically

MOC 50432 Excel 2007 Programming Using Visual Basic for Applications

Course Outline (cont'd)

VI. Working with Forms and Controls
This module explains how to create a UserForm. It describes several controls and control properties. Creating event procedures for controls is also covered.

A. Understanding UserForms
B. Using the Toolbox
C. Working with UserForm Properties, Events, and Methods
D. Understanding Controls
E. Setting Control Properties in the Properties Window
F. Working with the Label Control
G. Working with the Text Box Control
H. Working with the Command Button Control
I. Working with the Combo Box Control
J. Working with the Frame Control
K. Working with Option Button Controls
L. Working with Control Appearance
M. Setting the Tab Order
N. Populating a Control
O. Adding Code to Controls
P. Launching a Form in Code

Lab: Working with Forms and Controls
Adding a Userform to a Project

- Using 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 Forms from Procedures

After completing this module, students will be able to:

- Create a Userform.
- Add controls to a Userform.
- Position, size, align, and group form controls.
- Set the tab order of controls on a form.
- Add list items to a Combo Box control.
- Create event procedures for controls and add code to set control properties.
- Write code to launch a Userform in a procedure.

VII. Working with the PivotTable Object
This module explains how to work with PivotTable objects using VBA code. It also shows how to assign a macro to the Quick Access toolbar.

A. Understanding PivotTables
B. Creating a PivotTable Using Worksheet Data
C. Working with the PivotTable Objects
D. Working with the PivotFields Collection
E. Assigning a Macro to the Quick Access Toolbar

Lab: Working with the PivotTable Object

- Creating a PivotTable Using Worksheet Data
- Working with the PivotTable Objects
- Working with the PivotFields Collection
- Review – Completing the Run Reports Form
- Assigning a Macro to the Quick Access Toolbar

After completing this module, students will be able to:

- Create a PivotTable using the Excel interface.
- Create a PivotTable in code.
- Assign Page, Row, Column, and Data fields for a PivotTable programmatically.
- Assign a macro to the Quick Access toolbar.

Due to the nature of this material, this document refers to numerous hardware and software products by their trade names. References to other companies and their products are for informational purposes only, and all trademarks are the properties of their respective companies. It is not the intent of ProTech Professional Technical Services, Inc. to use any of these names generically

MOC 50432 Excel 2007 Programming Using Visual Basic for Applications

Course Outline (cont'd)

VIII. Debugging Code

This module explains the debugging tools in the Visual Basic Editor and their use. It also describes break mode, how to set breakpoints, and how to determine the value of code expressions while in break mode.

A. Understanding Errors
B. Using Debugging Tools
C. Setting Breakpoints
D. Stepping through Code
E. Using Break Mode during Run mode
F. Determining the Value of Expressions

Lab: Debugging Code

- Setting Breakpoints
- Stepping through Code
- Determining the Value of Expressions

After completing this module, students will be able to:

- Use Break mode and set a breakpoint on a line of code.
- Step through code one line at a time.
- Use a variety of tools to determine the value of expressions while in Break mode.

IX. Handling Errors

This module explains how to trap errors and write an error handling routine. It also describes the Err object and how to set VBA error trapping options.

A. Understanding Error Handling
B. Understanding VBA's Error Trapping Options
C. Trapping Errors with the On Error Statement
D. Understanding the Err Object
E. Writing an Error-Handling Routine
F. Working with Inline Error Handling

Lab: Handling Errors

- Understanding VBA's Error Trapping Option
- Writing an Error-Handling Routine
- Working with Inline Error Handling

After completing this module, students will be able to:

- Set VBA error trapping options for debugging.
- Write code to trap errors with the **On Error** statement.
- Write an error-handling routine.
- Write code to handle errors inline.

Due to the nature of this material, this document refers to numerous hardware and software products by their trade names. References to other companies and their products are for informational purposes only, and all trademarks are the properties of their respective companies. It is not the intent of ProTech Professional Technical Services, Inc. to use any of these names generically