

# Microsoft Excel - Macros and VBA

GoSkills online course syllabus

Sunday, June 23, 2024

<b>Skill level</b>	<b>Lessons</b>	<b>Accredited by</b>
Intermediate	41	CPD
<b>Pre-requisites</b>	<b>Versions supported</b>	<b>Video duration</b>
<a href="#">Excel - Basic</a>	2016, 2019, 2021, 365	3h 47m
<b>Estimated study time</b>	<b>Instructor</b>	
20h 30m for all materials	Ken Puls	

## Essential Background

---

- 1** **Macros vs VBA - What's the Difference?**  
A brief discussion about the differences between macros and VBA, and how this course covers both.
- 2** **Setting up the Macro Environment**  
Entry level steps to allow the user to record macros in Excel.

## Getting Started with Macros in Excel

---

- 3** **Creating Your First Macro**  
Recording a simple Macro in Excel.
- 4** **File Types & Saving**  
A discussion on which files types support macros, and why it is critical to save your work before moving forward.
- 5** **Running Macros: Using the Macro Dialog**  
How to run a macro from the Macro dialog.
- 6** **Running Macros: Using Keyboard Shortcuts**  
How to run a macro from a keyboard shortcut.
- 7** **Running Macros: Using Buttons**  
How to run your macro from a worksheet button.

## Getting Started with VBA

---

## 8 Navigating the Visual Basic Editor

Meet the Visual Basic Editor (VBE) - your coding studio.

## 9 VBA Objects & the Object Model

A discussion of the different items you'll encounter as you learn to code, and how they relate to each other.

## 10 Excel's VBA Object Model

A visual view of a portion of Excel's object model.

## 11 Where Do I Put My Code?

Examining the different code containers and where you should place your code.

## 12 Understanding Code: Macro Syntax

Looking at the different keywords and structure that make up a valid macro.

## 13 Cleaning up Recorded Code

Editing the previously recorded code in order to remove unnecessary objects.

## Debugging: Entry Level

---

### 14 Step vs Run

How to step through a macro line by line in order to aid in debugging.

### 15 Using Breakpoints

How to set and use breakpoints during code development and debugging.

## Variables

---

### 16 What are Variables?

A discussion of what variables are, and what they do for us when coding.

### 17 Creating Variables

Setting up variable dimensions and ensuring the code does so in the correct location.

**18** **Setting Variables**  
How to assign strings, values or objects to variables and use them in your code.

**19** **Explicit vs Implicit Variables**  
Why forgetting to set a simple flag can burn you in the long term.

## Debugging - Advanced

---

**20** **Using the Locals Window**  
Working with the locals window to help you debug and explore the object model.

**21** **Using the Immediate Window**  
Exploring the benefits of the Immediate window for logging and querying, as well as writing when needed.

**22** **Using the Watch Window**  
How to use the Watch window to break code execution when variable conditions are met.

**23** **Using the Stop Keyword**  
How the Stop keyword can be used during code development similar to a breakpoint.

## Coding Constructs

---

**24** **With Blocks**  
How "With" blocks can tighten up your code and ensure your code targets the objects you expect.

**25** **Logic Tests: If Then Else**  
Implementing If/Then choices in VBA.

**26** **Logic Tests: Select Case**  
Understanding how the Select Case construct adds another logic test to your coding arsenal.

**27** **Loops: Basic Looping with Do Loops**  
Basic looping including counting iterations and exiting.

**28** **Loops: Looping under Conditions with Do While/Until Loops**  
More advanced looping by looping while or until a certain condition is met.

## 29 Loops: Looping X Iterations with For X to Y Loops

Running a loop a set number of times.

## 30 Loops: Looping Through Collections with For Each X in Y Loops

Using a For Each loop to cycle through each object in a collection such as each worksheet in a workbook's worksheets.

## 31 Calling Other Macros

Setting up a master macro allowing you to call other macros from a single source.

# User Feedback and Inputs

---

## 32 Creating VBA Message Boxes

How to provide feedback to your user via the VBA MsgBox object.

## 33 Collecting Feedback from a VBA MsgBox

Identifying which button the user clicked when presented with a MsgBox in order to use their response in our code.

## 34 Collecting Feedback from a VBA InputBox

Working with the VBA InputBox to prompt the user to enter information and capture it for later use.

## 35 Forcing User Input

Forcing users to enter data when requested.

# Error Handling Techniques

---

## 36 Error Types

How to trigger various errors in VBA, and what they mean.

## 37 Trapping and Handling Errors

How to set up an error trap in VBA to handle errors.

## 38 Building Error Handlers

Setting up an error handling section for your macro.

# User Defined Functions (UDFs)

---

**39** User Defined Function (UDF) Syntax  
The syntax signature for a UDF and how it differs from a standard subroutine.

**40** Creating User Defined Functions (UDFs)  
Creating a UDF to return the user name.

**41** Calling a User Defined Function (UDF)  
Calling a UDF from a worksheet and from VBA.

[Go to GoSkills.com](https://www.goskills.com)