

What is VBA?

Visual Basic for Applications (VBA) is the underlying programming language behind Excel macros and is used to adapt and extend the capability of Excel. At a simple level, tasks which are repetitive and lengthy can be recorded and then replayed in a matter of moments. You can also use VBA to write your own functions (User Defined Functions) to simplify more complex formulae.

Who is this for?

You're already using Excel regularly and will have a good working knowledge of around 60% to 70% of its features. You'll also have many repetitive tasks which could be automated; for example adding standard headers and footers to spreadsheets, entering specific data into cells, or formatting your spreadsheet to a set standard.

What does it cover?

This course recaps and builds upon the macro elements of the Advanced course and introduces the Visual Basic for Applications (VBA) language. You'll learn how to record and replay macros, understand how the recorded macro relates to the task being done, and carry out simple editing of the macro. You'll also learn what to do to test the macro, watch it work, and understand where errors can creep in.

How long is it?

One day as a group, or can be split into several sessions if required for individuals.

Where is it run?

On your site, at home, or via one of the venues we use.

I'm interested ...

Just email us on enquiries@whitehorsect.co.uk or visit www.whitehorsect.co.uk to get in touch.

Course content

Overview of VBA

What is VBA?

What can it do for me?

VBA Jargon - Understanding workbooks and VBA projects

Being repetitive – recording and running macros

Recording a few useful macros

- ♦ *Relative versus absolute recording*

Running a macro

Assigning a macro to the Quick access toolbar

Introducing the VBA editor – the IDE

The Integrated Development environment (IDE) explained

Menu & toolbar overview

The Project Explorer

Where does the code go?

- ♦ *Introducing Code Modules*
- ♦ *Naming your modules*

Watch your Macros operate

- ♦ *Tracing through code*

Responding to your User

Getting Input through input boxes

Feedback through Message Boxes

Variables & Constants dissected

Why use variables?

Creating, naming and assigning values to variables

- ♦ *Using Option Explicit*

Why use constants?

Creating and using constants

Controlling the flow ...

Making decisions with

- ♦ *IF...THEN...ELSE*

Going through the loop

- ♦ *The FOR...NEXT*
- ♦ *The WHILE loops*

When things go wrong...

Setting breakpoints

Watching variables change

- ♦ *Setting up the Watch Window*
- ♦ *Adding watches to variables*

More overleaf...

The Excel Object Browser

Methods, properties & events

Reusing your code - subroutines

What are Subroutines?

Creating & naming subroutines

Calling a subroutine

Getting data into the subroutine - basic parameter passing

Reusing your code - functions

What are User Defined Functions (UDF)?

- ♦ *Advantages of a UDF*

Creating & declaring Functions

Getting data into the function – Basic parameter passing

Introduction to creating your own Excel user interface

Creating a custom user form

Understanding events

Adding a push button and text box

- ♦ *Adding VBA code to the push button's event*
- ♦ *Sending the text box contents to a spreadsheet*
- ♦ *Reading text from a sheet to the text box*

Adding a combo box

- ♦ *Setting the list items from a spreadsheet range*
- ♦ *Picking the selected item*
- ♦ *Sending the selected item to a spreadsheet*