

The book was found

Visual Basic For Applications (VBA) 2013 Quick Reference Guide: Introduction (Cheat Sheet Of Instructions, Tips & Examples - Laminated)

Visual Basic® for Applications 2003 Introduction

Visual Basic for Applications (VBA) is a programming language built into Word, Excel, Access, and other Microsoft applications. It can be automatically inserted into your programs through a set of instructions, commonly referred to as a code. VBA instructions are normally entered into a code window after each application Visual Basic object. Only Word and Excel have a menu for displaying the code windows for each code.

Showing the Developer Tab

Always display the DEVELOPER tab in the ribbon task pane to access VBA code and related security settings.

1. Choose **FILE > OPTIONS > DEVELOPER** ribbon.
2. Choose **DEVELOPER** ribbon.

Setting Macro Security

1. Choose **FILE > OPTIONS > SECURITY** ribbon.
2. Choose one of the Macro Security levels (Basic, Custom, or High).

Tip: Use the Enable All Macro option when entering on your own code.

Create VBA Code

1. Press **ALT+F11** to open the **VBA Editor** window.
2. Open the left-hand Project Explorer pane, if necessary, with **VIEW > PROJECT EXPLORER** or **CTRL+R**. There are 2 main types of code windows: Document, Module, and User. Document windows contain Office application documents. Module and User Form windows can be created for customized programming. VBA code is normally saved in a Module window. Tip: Press **ALT+F11** to toggle between the Project Explorer and the VBA Editor.

Project Explorer Names

1. Double-click the **Project Explorer** name in Project Explorer. If necessary, type in the following label to distinguish the folder's name:
 - If the code window is from a Word document, the Project Name and document Name (Module).
2. Choose **PROJECT > PROJECT** ribbon.
 - a. Enter a NAME (in Code Name Label).
 - b. Select **NO** for the value in Subroutine, or **FUNCTION** to create a function. Choose **OK**.
3. Enter the code that displays **Hi!** (see **Saving Code**).

Tip: Save your writing within a folder in Word by saving Macro within that saving the code window from its default location to your code.

Deleting a Code Window

1. Right-click its name in Project Explorer and select **DELETE** (remove window).
2. Click **NO** from the pop-up message asking you to enter the code for the subprogram.

Code Name Rules

- Code names must begin with a letter in Word and Excel but is good practice for all Office applications.
- Subsequent characters should only be letters, numbers, or underscore characters.
- Code practices include: no spaces, no file extensions, and subroutines and subroutines for spaces.
- Word names that are the same as cell references in Word or worksheet VBA formulas such as **DATE1**.

Important Keyboard Shortcuts

Shift+Alt – Shifts right when editing opening or closing brackets and the AutoCorrect pane will be displayed.

Ctrl+Break – Stops running code. This key combination is particularly useful in testing code.

Shift+F2 – Displays the VBA Editor Object Browser dialog box of an object, method or property selected in code windows.

Alt+R – Displays a list of macros in Word or Word.

Project Explorer Names

- Inserted modules or user forms are named Module1, UserForm1, UserForm2, etc.
- Add: Project name + Workbook name + Personal code. Code window name + Worksheet name.
- Word: Project name + Normal or Word for name. Code window name + Worksheet.



- PowerPoint: Project name + PowerPoint for the name. Code window name + Module1 (1, 2, 3, etc.).
- Access: Project name + Database file name. Code window name + Name of form or report.

Saving Code

Code can only be saved by using the Office application Mac is stored in. Default file types for Word, Word or PowerPoint (e.g., *.doc, *.docx) cannot contain macros, therefore in these programs:

1. Choose **FILE > SAVE** or press **CTRL+S**.
2. If "Macro-enabled" warning message pops up (Word) No.
3. Change **SAVE AS TYPE** to **MACRO-ENABLED** (*.docm, *.docx, *.pptm).
4. Choose location, specify **FILE NAME**, then **SAVE**.

Saving Global Macros

When entering a macro in Word or Excel you can choose to save your work into the current document or globally. To save a macro globally, you must globally choose the following from the **FILE > MACRO** drop-down list to the **FILE > MACRO > GLOBAL** dialog box:

- In Word choose **FILE > MACRO > GLOBAL > YES**.
- In Word choose **FILE > MACRO > YES** (without dialog).

VBA code can also be saved globally if it is manually entered into the **GLOBAL** project in Word or the **PERSONAL** XLST project in Excel.

VBA Editor Tips

1. The Project Explorer pane is best used to navigate between Code windows.
2. Use the Code window's Procedure drop list (see **Navigating Code Window Lists**) to navigate between subroutines in a long Code window.
3. Click the **Project Explorer** list box (see **Project Explorer**) to display the list of subroutines in the project and objects grouped by folders. Making this change is an easy Office application with all Office applications.
4. Close all windows and press **CTRL+Q** to display the window while writing code or before.
5. To directly enter a macro in Word or Word press **ALT+F11**, select the macro name, then **CTRL+Q**.
6. Press **CTRL+Q** to close the VBA Editor and return to the Office application.

Password Protect Code

1. Right-click your Project name or one of its components in the Project Explorer pane.
2. Select **PROJECT PROPERTIES** or **VBAPROJECT PROPERTIES**.
3. Check **PROTECTION** for Add your password to both the **FACTORY** and **COMPILED FACTORY** boxes. This restricts code editing.
4. Click **LOCK PROTECT FOR VIEWING** to also prevent code viewing.
5. Note that the **GENERAL** tab can be used to change the Project name and description.

Understanding the VBA Language

VBA code performs tasks in any language. If the structure of English sentences can be compared, Microsoft Office VBA code that the syntax of VBA or even can be applied to. In Microsoft Office, each code window, or document, contains a single task (Method) or change the appearance (Property) of a single object. As an easy reference, you can use a phrase that you enter immediately as a phrase (Method) or (Property) to perform an available or function.

Object Each Office application is divided into Objects that can be manipulated. Subroutines, methods, and cells. Objects can be changed by the application either as small as single characters.

Method Methods are operations that can be done on an Object. In Word a Selection (Object) of words can be typed (Method) as an Object (Object) as a Label (Object) from code (Property). In Access a Property (Object) can be changed (Method) for a list (Object) (Property).

Events Each Object has associated events. Events happen to Objects and are used as triggers to cause code execution. When an event (Object) is Selected (Event) as a Word document (Object) to Close (Event) a message can be programmed to pop up.

- To open a file of any type to reveal the VBA Editor Properties and Events. Select the Object (Object) in the Project Explorer list box. Select the Object (Object) in the Project Explorer list box. Select the Object (Object) in the Project Explorer list box. Select the Object (Object) in the Project Explorer list box.

Using Code Window Lists

Object list Procedure list

Object	Procedure
Project	Project

The Object list and the Procedure list are at the top of any Word code window, i.e., any Word document. When an event (Object) is Selected (Event) as a Word document (Object) to Close (Event) a message can be programmed to pop up.

1. Select an appropriate item from the Object list.

2. Select an appropriate item from the Procedure list.

3. Write code into the newly created subroutine. For example, open an Access Word code window. Select the name of a control, e.g., Text1, from the Object list and clicking down from its Procedure list. A new subroutine is created and the code is added. To type text, select a control, e.g., Text1, from the Object list and clicking down from its Procedure list. To type text, select a control, e.g., Text1, from the Object list and clicking down from its Procedure list. To type text, select a control, e.g., Text1, from the Object list and clicking down from its Procedure list.

Construct a Code Subroutine

1. Select an Object (e.g., Word, Range (VBA), Select) Object > Name, Method > Select.
2. Change the text value to select.
3. Select the Object, Method > Select, or Object > Name, Method > Select, or Object > Name, Method > Select, or Object > Name, Method > Select, or Object > Name



Synopsis

Four page laminated quick reference guide that will open the world of Microsoft Office automation in Word, Excel, PowerPoint and Access. You will learn how to modify macros, customize functions and create your own programming. Written with Beezix's trademark focus on clarity, accuracy, and the user's perspective, this guide will be a valuable resource for composing Visual Basic for Applications code. Geared toward the intermediate to advanced Microsoft Office 2013 user. The following topics are covered: Getting Started: Showing the Developer Tab; Setting Macro Security; Opening a Code Window; Deleting a Code Window; Saving Code; Project Explorer Names; Code Name Rules; VBA Editor tips; Password Protect Code & Understanding the VBA Language. Create VBA Subroutines: Useful Code Events; Navigating Code Window Lists: Object List Items and Procedure List Items. Organizing Code: Variables and Constants for Efficiency; Good Practices with Variables; Sophistication with VBA Functions: Useful VBA Function Categories; Decision Power with IF and SELECT & Eliminate Repeats with FOR and DO. Other Topics: Reach Out to Users; Troubleshooting & VBA Code Tips. A wide variety of examples provided for the following: Keyboard Shortcuts; Constructing a Code Statement; Simple Code; Variables; Code Decisions; Eliminating Repeats & Interacting with Users.

Book Information

Pamphlet: 4 pages

Publisher: Beezix Inc. (July 9, 2014)

Language: English

ISBN-10: 1939791332

ISBN-13: 978-1939791337

Product Dimensions: 11 x 9 x 0.2 inches

Shipping Weight: 0.3 ounces (View shipping rates and policies)

Average Customer Review: 4.2 out of 5 stars Â Â See all reviews Â (4 customer reviews)

Best Sellers Rank: #405,369 in Books (See Top 100 in Books) #115 in Â Books > Computers & Technology > Programming > Languages & Tools > Visual Basic #498 in Â Books > Computers & Technology > Programming > Microsoft Programming #813 in Â Books > Computers & Technology > Programming > Introductory & Beginning

Customer Reviews

These are handy to have for homework and learning or for a test if you have an open book type.

They typically are not deep in their subject but will give you the basic reference and more to refresh

your memory or get you started.

OK I guess if your a novice.

This was what I needed for a class

Handy for the price.

[Download to continue reading...](#)

Visual Basic for Applications (VBA) 2013 Quick Reference Guide: Introduction (Cheat Sheet of Instructions, Tips & Examples - Laminated) Microsoft Project 2013 Quick Reference Guide: Creating a Basic Project (Cheat Sheet of Instructions, Tips & Shortcuts - Laminated Card) Microsoft SharePoint 2013 Quick Reference Guide: Introduction (Cheat Sheet of Instructions & Tips for On-Premises Environments - Laminated Card) Microsoft PowerPoint 2013 Introduction Quick Reference Guide (Cheat Sheet of Instructions, Tips & Shortcuts - Laminated Card) Microsoft Excel 2013 Data Analysis with Tables Quick Reference Guide (Cheat Sheet of Instructions, Tips & Shortcuts - Laminated Card) Microsoft Project 2013 Quick Reference Guide: Managing Complexity (Cheat Sheet of Instructions, Tips & Shortcuts - Laminated Card) Microsoft Project 2016 Quick Reference Guide Creating a Basic Project - Windows Version (Cheat Sheet of Instructions, Tips & Shortcuts - Laminated Card) Microsoft Project 2010 Quick Reference Guide: Creating a Basic Project (Cheat Sheet of Instructions, Tips & Shortcuts - Laminated Card) Microsoft Word 2013 Intermediate Quick Reference: Paragraphs, Tabs, Columns, Tables & Pictures (Cheat Sheet of Instructions, Tips & Shortcuts - Laminated Card) Mac OS X El Capitan Introduction Quick Reference Guide (Cheat Sheet of Instructions, Tips & Shortcuts - Laminated Guide) Mac OS X Yosemite Introduction Quick Reference Guide (Cheat Sheet of Instructions, Tips & Shortcuts - Laminated Guide) Mac OS X Mavericks Introduction Quick Reference Guide (Cheat Sheet of Instructions, Tips & Shortcuts - Laminated Guide) Windows 10 Introduction Quick Reference Guide (Cheat Sheet of Instructions, Tips & Shortcuts - Laminated) Mac OS X Lion Introduction Quick Reference Guide (Cheat Sheet of Instructions, Tips & Shortcuts - Laminated Card) Windows 8.1 Update Quick Reference Guide: Introduction (Cheat Sheet of Instructions, Tips & Shortcuts - Laminated) Windows 8.1 Quick Reference Guide: Introduction (Cheat Sheet of Instructions, Tips & Shortcuts - Laminated) Microsoft Windows XP Introduction Quick Reference Guide (Cheat Sheet of Instructions, Tips & Shortcuts - Laminated Card) Microsoft SharePoint 2010 Quick Reference Guide: Introduction (Cheat Sheet of Instructions, Tips & Shortcuts - Laminated Card) Microsoft Excel 2016 Introduction

Quick Reference Guide - Windows Version (Cheat Sheet of Instructions, Tips & Shortcuts -
Laminated Card) Windows 7 Quick Reference Guide (Cheat Sheet of Instructions, Tips & Shortcuts
- Laminated Card)

[Dmca](#)