Visual Basic 2008 Made Easy

Dr. Liew Voon Kiong
Disclaimer

Visual Basic 2008 ® Made Easy- A complete tutorial for beginners is an independent publication and is not affiliated with, nor has it been authorized, sponsored, or otherwise approved by Microsoft Corporation.

Trademarks

Microsoft, Visual Basic, Excel and Windows are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries.

Liability

The purpose of this book is to provide a basic guide for people interested in Visual Basic® programming. Although every effort and care has been taken to make the information as accurate as possible, the author shall not be liable for any error, harm or damage arising from using the instructions given in this book.
Acknowledgement

I would like to express my sincere gratitude to many people who have made their contributions in one way or another to the successful publication of this book. My special thanks go to my children Xiang, Yi and Xun. My daughter Xiang edited this book while my sons Yi and Xun contributed their ideas and even wrote some of the sample programs for this book. I would also like to appreciate the support provided by my beloved wife Kim Huang and my youngest daughter Yuan. I would also like to thank the millions of visitors to my Visual Basic Tutorial website at www.vbtutor.net, especially those who contributed their comments, for their support and encouragement.

About the Author

Dr. Liew Voon Kiong holds a bachelor's degree in Mathematics, a master's degree in management and a doctoral degree in Business Administration. He has been involved in programming for more than 15 years. He created the popular online Visual Basic Tutorial at www.vbtutor.net in 1996 and since then the web site has attracted millions of visitors and it is one of the top searched Visual Basic Tutorial websites in many search engines including Google. To provide more support for the Visual Basic hobbyists, he has written this book based on the Visual Basic tutorial. He is also the author of Visual Basic 6 Made Easy, Visual Basic 2010 Made Easy and Excel VBA Made Easy.
# TABLE OF CONTENTS

Chapter 1 Introduction 10

1.1 A brief description of Visual Basic 2008 10
1.2 The VB 2008 Integrated Development Environment 10

Chapter 2 Working with Controls 15

2.1 Using Text Box 16
2.2 Using List Box 16

Chapter 3 Working with Control Properties 18

Chapter 4 Object Oriented Programming 22

4.1 Encapsulation 22
4.2 Inheritance 22
4.3 Polymorphism 23
   Example 4.1 BMI Class 25

Chapter 5 Writing the Code 28

5.1 The event Procedure 28
5.2 Writing the code 29
   Example 5.1 29
   Example 5.2 30

Chapter 6 Managing VB2008 Data 32
6.1 Visual Basic 2008 Data Types
6.1.1 Numeric Data Types
6.1.2 Non-numeric Data Types
6.1.3 Suffixes for Literals

6.2 Managing Variables
6.2.1 Variable Names

6.2.2 Declaring Variables
Example 6.1
6.2.3 Assigning Values to Variables
Example 6.2 Declaring and assigning initial values to variables
Example 6.3 Declaring variables then assign values to them

6.3 Constants
6.3.1 Declaring a Constant
Example 6.4

Chapter 7 Mathematical Operations
Example 7.1 Basic Arithmetic Operations
Example 7.2 Pythagoras Theorem
Example 7.3 BMI Calculator

Chapter 8 String Manipulation
8.1 String Manipulation Using + and & signs.
Example 8.1 Text Concatenation
Example 8.2 String Concatenation

8.2 String Manipulation Using VB2008 Built-in Functions
8.2.1 The Len Function
Example 8.3 Compute the Length of Text
8.2.2 The Right Function
Example 8.4 Extracting a Phrase with the Right Function
8.2.3 The Left Function

Chapter 9 Controlling Program Flow
9.1 Conditional Operators

9.2 Logical Operators

9.3 Using If control structure with Comparison Operators
9.3.1 If...Then Statement
16.2 Formatting Date and time using user-defined formats 85
  Example 16.2 Formatting Dates and Time with User-Defined Formats 86

Chapter 17 Using the Check Box 87
  Example 17.1 Shopping Cart 88
  Example 17.2 Another Shopping Cart 89
  Example 17.3 Formatting Fonts 90

Chapter 18 Using the Radio Button 92
  Example 18.1 Color Selection 92
  Example 18.2 Shopping Cart 93

Chapter 19 Creating a Web Browser 96

Chapter 20 Errors Handling 99
  20.1 Introduction 99
  20.2 Using On Error GoTo Syntax 99
    Example 20.1 Division by Zero 100
  20.3 Errors Handling using Try...Catch...End Try Structure 101
    Example 20.2 Data Types Mismatch Error 102

Figure 20-2: The Output Window

Chapter 21 Reading and Writing Files 103
  21.1 Introduction 104
  21.2 Reading a Text File 104
  21.3 Writing to a Text File 108

Chapter 22 Creating and Managing Graphics 111
  22.1 Introduction 111
  22.2 Creating the Graphics Object 111
  22.3 Creating a Pen 112
  22.4 Drawing a Line 113
    Example 22.1 Drawing a Straight Line 113
  22.5 Creating a Rectangle 114
    Example 22.2 Drawing a Rectangle 115
  22.6 Customizing Line Style of the Pen Object 116
    Example 22.3 Drawing a Rectangle with Dotted Line 116
Chapter 1 Introduction

❖ A brief description of Visual Basic 2008
❖ Getting to know the Visual Basic 2008 Integrated Development Environment

1.1 A brief description of Visual Basic 2008

Visual Basic 2008 is the latest version of Visual Basic launched by Microsoft in the year 2008. It is almost like Visual Basic 2005, but it has added many new features. Visual Basic has gone through many phases of development since the days of BASIC that was built for DOS. BASIC stands for Beginners' All-purpose Symbolic Instruction Code. The program code in Visual Basic resembles the English language. Different software companies had produced many different versions of BASIC for DOS, such as Microsoft QBASIC, QUICKBASIC, GWBASIC, and IBM BASICA and more. Then, Microsoft launched the first graphical BASIC which was known as Visual Basic Version1 in 1991. It is GUI based and especially developed for MS window. Since then the DOS versions of BASIC were slowly phased out and almost completely replaced by Visual Basic.

Visual Basic was initially a functional or procedural programming language until the popular Visual Basic 6. Then, Microsoft decided to make Visual Basic into more powerful object-oriented programming language, Visual Basic 2005 was launched with that purpose in mind. Visual Basic 2005 is an object-oriented programming language and it was to be taken over by Visual Basic 2008. Visual Basic 2008 is a full-fledged Object-Oriented Programming (OOP) Language, so it has caught up with other OOP languages such as C++, Java, C# and others. However, you do not have to know OOP to learn VB2008. In fact, if you are familiar with Visual Basic 6, you can learn VB2008 effortlessly because the syntax and interface are almost similar. Visual Basic 2008 Express Edition is available free for download from the Microsoft site.

1.2 The VB 2008 Integrated Development Environment
When you launch VB2008 Express, the Integrated Development Environment will be presented to you, as shown in Figure 1-1.

Figure 1-1 The VB2008 IDE

The IDE consists of a few panes, namely:

- The Recent Projects Pane- it shows the list of projects that have been created by you recently.
- The Getting Started Pane- It provides some helpful tips to quickly develop your applications
- The VB Express Headlines pane- It provides latest online news about Visual Basic 2008 Express. It will announce new releases and updates.

To start creating your first application, you need to click on File on the menu bar and select New Project. The VB2008 New Project dialog box will appear, as shown in Figure 1-2
The dialog box offers you five types of projects that we can create. Since we are keen only in creating a Windows application, we select Windows Forms Application. At the bottom of this dialog box, change the default project name `WindowsApplication1` to some other name, for example, `MyFirstProgram`. After renaming the project, click OK to continue. A new Form will appear, as shown in Figure 1-3. It consists of an empty form, the common controls toolbox, the solution explorer, and the Properties Window.
Now let us create your first program. First, drag one common button into the form and change its default name to calculate, as shown in Figure 1-4.

Next, click on the calculate button and enter the following code at the source code window as shown in Figure 1-5.
Private Sub Button1_Click (ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button1.Click
    Dim num1, num2, sum As Single
    num1 = 100
    num2 = 200
    sum = num1 + num2
    MsgBox ("The Sum of" & num1 & " and " & num2 & " is " & sum)
End Sub

Now run your first application! And you can see the follow message box showing the sum of two numbers.

Figure 1-5 The Code window
Figure 1-6 The Output Window
Chapter 2 Working with Controls

❖ Getting to know Visual Basic 2008 controls
❖ Getting to know the Control Properties.

The Visual Basic 2008 Common Control Toolbox consists of all the controls essential for developing a VISUAL BASIC 2008 application. Controls in VB2008 are useful tools that can be placed in the form to perform various tasks. They are used to create many kinds of Windows applications. The diagram on the right is the Toolbox that contains the controls of VB2008. They are categorized into Common Controls, Containers, Menus, Toolbars, Data, Components, Printings and Dialogs. Now, we will focus on the common controls. Some of the most used common controls are Button, Label, ComboBox, ListBox, PictureBox, TextBox and more, as shown in Figure 2-1.

To insert a control into your form, you just need to drag the control and drop it into the form. You can reposition and resize it as you like. Let us examine a few programs that made use of Button, Label, TextBox, ListBox and PictureBox. You do not have to worry so much about the code because I will explain the program syntax as you progress to later Chapters.
2.1 Using Text Box

In this program, you insert two textboxes, three labels and one button. The two textboxes are for the users to enter two numbers, one label is to display the multiplication operator and the other label is to display the equal sign. The last label is to display the answer. The run time interface is shown in Figure 2-2

![Image of the Multiplication Program](image)

Figure 2-2 The Multiplication Program

The Code

```vbnet
Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button1.Click
    Dim num1, num2, product As Single
    num1 = TextBox1.Text
    num2 = TextBox2.Text
    product = num1 * num2
    Label3.Text = product
End Sub
```

2.2 Using List Box

This program will add one item at a time as the user enters an item into the text box and click the Add button. In this program, you insert a textbox and a list box into the Form. The function of the textbox is to let the user enter an item one at a time and
add it to the list box. The method to add an item to the ListBox is Add. The output interface is shown in Figure 2-3.

![Figure 2-3: The Add Items Program](image)

**The Code**

```vbnet
Class Frm1

Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button1.Click
    Dim item As String
    item = TextBox1.Text

    'To add items to a listbox
    ListBox1.Items.Add(item)

End Sub
End Class
```
Chapter 3 Working with Control Properties

❖ Setting the properties of the controls

3.1 Setting Control Properties in the Properties Window
Before writing an event procedure for the control to response to a user’s input, you have to set certain properties for the control to customize its appearance and the way it will work with the event procedure. You can set the properties of the controls in the properties window at design time or you can set the properties using code.

Figure 3-1 on the right is a typical properties window for a form. The title of the form is defined by the Text property and its default name is Form 1. To change the form’s title to any name that you like, simply click in the box on the right of the Text property and type in the new name, in this example, the title is Multiplication. Notice that this title will appear on top of the windows. In the properties window, the item appears at the top part is the object currently selected (in Figure 3.1, the object selected is Form1). At the bottom part, the items listed in the left column represent the names of various properties associated with the selected object while the items listed in the right column represent the states of the properties. Properties can be set by

Figure 3-1: The Properties Window
highlighting the items in the right column
then change them by typing or selecting
the options available. You may also alter other properties of the form such as font, location, size, foreground color, background color, MaximizeBox, MinimizeBox and more.
For example, when you select background color, the dialog box for color selection will appear, as shown in Figure 3-2. You can then select any color by clicking on one of the colors.

Figure 3-2: Color Selection

3.2 Setting Control Properties using Code

You can also change the properties of the object at run time to give special effects such as change of color, change of shape, animation effect and more. For example,
the following code will change the form color to yellow every time the form is loaded. VB2008 uses RGB(Red, Green, Blue) to determine the colors. The RGB code for yellow is 255,255,0. Me in the code refer to the current form and Backcolor is the property of the form's background color. The formula to assign the RGB color to the form is Color.FromArgb(RGB code).

Private Sub Form1_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load
    Me.BackColor = Color.FromArgb(255, 255, 0)
End Sub

You may also use the following procedure to assign the color at run time.

Private Sub Form1_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load
    Me.BackColor = Color.Yellow
End Sub

Both procedures above will load the form with a yellow background, as shown in Figure 3-3.

![Figure 3-3: The form with yellow background](image)

Here are some of the common colors and the corresponding RGB codes. You can always experiment with other combinations but remember the maximum number for each color is 255 and the minimum number is 0. The table below shows some of the common colors with their corresponding codes.
Table 3-1: Some common colors and their corresponding RGB codes.

<table>
<thead>
<tr>
<th>Color</th>
<th>RGB code</th>
<th>Color</th>
<th>RGB code</th>
<th>Color</th>
<th>RGB Code</th>
</tr>
</thead>
<tbody>
<tr>
<td>Red</td>
<td>255, 0, 0</td>
<td>Yellow</td>
<td>255, 255, 0</td>
<td>Orange</td>
<td>255, 105, 0</td>
</tr>
<tr>
<td>Green</td>
<td>0, 255, 0</td>
<td>Light Blue</td>
<td>0, 255, 255</td>
<td>Black</td>
<td>0, 0, 0</td>
</tr>
<tr>
<td>Blue</td>
<td>0, 0, 255</td>
<td>Blue</td>
<td>255, 0, 255</td>
<td>White</td>
<td>255, 255, 255</td>
</tr>
</tbody>
</table>

The following is a program that allows the user to enter the RGB codes into three different Textboxes and when he/she clicks the display color button, the background color of the form changes according to the RGB code. So, this program allows the user to change the color properties of the form at run time.

The code

Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button1.Click
    Dim rgb1, rgb2, rgb3 As Integer
    rgb1 = TextBox1.Text
    rgb2 = TextBox2.Text
    rgb3 = TextBox3.Text
    Me.BackColor = Color.FromArgb(rgb1, rgb2, rgb3)
End Sub

The output interface is shown in Figure 3-4
Figure 3-4: The RGB Program